



# Basic Integration Package (BIP)

Release Notes - ARCMW

Status: Release

## Copyright

---

The data in this document may not be altered or amended without special notification from ETAS GmbH. ETAS GmbH undertakes no further obligation in relation to this document. The software described in it can only be used if the customer is in possession of a general license agreement or single license. Using and copying is only allowed in concurrence with the specifications stipulated in the contract.

Under no circumstances may any part of this document be copied, reproduced, transmitted, stored in a retrieval system, or translated into another language without the express written permission of ETAS GmbH.

© **Copyright** 2022 ETAS GmbH, Stuttgart

The names and designations used in this document are trademarks or brands belonging to the respective owners.

---

## Contents

1	Introduction .....	5
1.1	Who Should Read This Document .....	5
1.2	How Will You Receive More Information .....	5
1.3	Intended Use .....	5
1.4	Rights of Use .....	5
1.4.1	Development License for ETAS Standard Software .....	5
1.4.2	Production License for ETAS Standard Software .....	5
2	Release Description .....	6
2.1	Release Information .....	6
2.2	Release Package .....	6
2.2.1	Features at a Glance .....	6
2.3	Folder Structure .....	10
2.4	Targets.....	12
2.4.1	Target Hardware.....	12
2.4.2	Restrictions.....	12
2.5	Tools.....	12
2.5.1	AUTOSAR Configuration Tools.....	12
2.5.2	Build Tools.....	13
2.5.3	Build options.....	13
2.5.4	RTE Command Line Option .....	13
2.6	Compatibility.....	13
2.7	Manual Modifications .....	14
2.8	Fixed Problems .....	14
2.9	Limitations.....	14
2.10	Known issues.....	14
2.11	Open-Source Software.....	14
3	Software Metrics .....	14
3.1.1	Memory Consumption.....	14
3.1.2	CPU Load .....	15
3.1.3	Stack Consumption .....	15
4	ETAS Contact Addresses .....	16

---

## Tables

Table 1 Release Information .....	6
Table 2 Release Package.....	6
Table 3 Feature of BIP .....	10
Table 4 Folder Structure.....	12
Table 5 Configuration Tool .....	13
Table 6 Build Tool.....	13
Table 7 Build options.....	13
Table 8 RTE options .....	13
Table 9 Manual Modifications.....	14
Table 10 Fixed Problems .....	14
Table 11 Limitations .....	14
Table 12 Known Issues .....	14

## 1 Introduction

---

This document provides release information for the ETAS BIP based on scope of work agreed in statement of work.

### 1.1 Who Should Read This Document

---

The ETAS RTA-CAR users who want to use the **Basic Integration Package (BIP)** as baseline to iterate their own projects with project specific requirements should read this document before use.

### 1.2 How Will You Receive More Information

---

For integration with your application, please refer to the integration guide mentioned in chapter 2.2.

### 1.3 Intended Use

---

The configuration and integration of the software, including integration code, is not fully ready for production immediately by itself. Customer shall not use these directly into production ECU without fully verification and validation.

You are hereby granted a non-transferable, non-sublicenseable, temporary right to use the preview software release for the aforementioned purpose only. You are prohibited from using the results generated during evaluation and/or testing in applications without the benefit of your own quality assurance testing, and are likewise barred from forwarding the said results to third parties for applications. In the event that you deploy the preview software for testing a series production product, you shall be responsible to verify the results with the use of series products in each case and with suitable scope.

You are prohibited from distributing, modifying and/or adapting, reverse assembling or reverse compiling the preview software, or to translate it by any other means. You are only entitled to make copies of the preview software as may be required for reasonable and sufficient evaluation and/or testing.

ETAS shall retain all rights and title in the preview software and documentation (including but not limited to copyright, patents, trademarks and other intellectual property rights).

### 1.4 Rights of Use

---

#### 1.4.1 Development License for ETAS Standard Software

---

In the case that a development license has not been purchased, Basic Integration Package (BIP) shall not be entitled to copy, modify, distribute, transfer and sublicense the Generated Code or Executable Image for commercial production purposes.

#### 1.4.2 Production License for ETAS Standard Software

---

In the case that a production license has not been purchased, Basic Integration Package (BIP) shall not be entitled to copy, modify, distribute, transfer and sublicense the Generated Code or Executable Image for commercial production purposes.

## 2 Release Description

### 2.1 Release Information

Release Type	<i>R1.0</i>
Release Version	ETAS_BIP_PT_ARCMW_11.1_CAL60S244AB_CAR921
Scope of Release	This release contains ETAS RTA-CAR configuration and MCAL software integration for features in section [2.2] with toolchains in section [2.6].
Delivery Method	This software is delivered with AdHoc Link. Folder structure of the delivery contents is described in <a href="#">folder structure section below</a>

Table 1 Release Information

### 2.2 Release Package

Filename	Description
ETAS_BIP_PT_ARCMW_11.1_CAL60S244AB_CAR921	Software of ETAS AUTOSAR Basic Integration Package (BIP)
.\Docs\IntegrationGuide	Contain Integration Guideline
.\Docs\OpenSourceSoftware	Contain open-source software scan report
.\Docs\ReleaseNote	Contain document describing the release
.\Docs\IntegrationTest*	Contain document describing BIP integration test report

\*These are continuous integration test according to SOW phases.

Table 2 Release Package

#### 2.2.1 Features at a Glance

BIP delivers the following AUTOSAR R4.x conformant integrated basic software features.

Functions	Description
Power Management (Startup & Shutdown)	<ol style="list-style-type: none"> <li>AUTOSAR EcuM flexible with BswM managed start-up and shutdown phases <ul style="list-style-type: none"> <li>Startup-I</li> <li>Startup-II</li> <li>RUN</li> <li>POST-RUN</li> <li>PRE-SHUTDOWN</li> <li>SHUTDOWN</li> </ul> </li> <li>EcuM measured timestamp for Startup and Shutdown.</li> <li>Generic interface RE_ExeMgr_GetWakeupSource for wakeup source (IO).</li> <li>User mode switch interfaces for RUN -&gt; APP_RUN, APP_RUN -&gt; POST-RUN.</li> <li>Support EcuM_OnGoOffOne with user flexible integration in EcuM_GoOffOne().</li> <li>Support EcuM_OnGoOffTwo with user flexible integration.</li> <li>TstM_PostRun() for shutdown test.</li> <li>NvM_Integration_WriteAll() for shutdown NvM store.</li> <li>RE_ExeMgr_ExecuteShutdown() for user actions right before shutdown.</li> <li>Ecu_SwitchOff() for user implementation specific shutoff power supply.</li> <li>UpperTest SWC for Power Management.</li> </ol>

Functions	Description
Wakeup & Sleep	<p><b>Wakeup:</b> Coordination between BswM and ExeMgr SWC to detect valid wakeup source and control startup state transition.</p> <p><b>Sleep:</b> Coordination between BswM, EcuM to detect No Communication and determine sleep process (MCU power off).</p>
CAN & COM	<ol style="list-style-type: none"> <li>1. Support DBC import with Cobra auto conf-gen.</li> <li>2. One channel of CAN communication, with baud rate of 500kbps, and both sending and receiving are in polling mode.</li> <li>3. BIP application message DBC file ETAS_CanCom.dbc, which contains 4 Standard CAN messages. Among them, 2 receive messages and 2 send messages, both of which are FULLCAN.</li> <li>4. Sending messages are triggered periodically.</li> <li>5. Unused bits are padded with 0xFF.</li> <li>6. Example of CompuMethod for receiving and sending signals.</li> <li>7. Example of enabling and disabling ComIPduGroup using BswM, which is divided into two ComIPduGroups according to the sending and receiving directions of application message.</li> <li>8. Application message test SWC (INP_SWC, OUTP_SWC) with sample test code.</li> </ol>
CanSM	<ol style="list-style-type: none"> <li>1. Support DBC import with Cobra auto conf-gen.</li> <li>2. BusOff: BusOff fast and slow recovery strategy and fault handling mechanism.</li> </ol>
CanNM	<ol style="list-style-type: none"> <li>3. Support DBC import with Cobra auto conf-gen.</li> <li>4. BIP network management implementation based on AUTOSAR.</li> <li>5. example of active network wakeup.</li> <li>6. network management message DBC file ETAS_CanNm.dbc, including 1 send and 1 receive NM message.</li> <li>7. Support using user-data interface function to receive and send 6-byte user data.</li> <li>8. Provides the NM state switching notification callback function and the function definition example of the NM message receiving notification callback function.</li> <li>9. Provide network management test SWC (NmUT) and sample test code to realize the definition of reading and writing userdata, requesting, releasing network and callback function.</li> </ol>

Functions	Description
Dcm	<ol style="list-style-type: none"> <li>1. Auto conf-gen from DEXT.</li> <li>2. Provide support for the following services 0x10 0x27 0x3E 0x11 0x28 0x85 0x22 0x2E 0x19 0x14 0x2F 0x31.</li> <li>3. 10 service supports Default/Programming/Extended session.</li> <li>4. 27 service supports security access group 1/5, 2/6 sub-services.</li> <li>5. 11 service supports Hard Reset and Soft Reset sub-services.</li> <li>6. 28 service supports enableRxAndTx and disableRxAndTx sub-services.</li> <li>7. 85 service supports DTC ON and DTC OFF sub-services.</li> <li>8. 19 service supports #1, #2, #4, #6, #A sub-services.</li> <li>9. UDS transmission RX/TX buffer size 255.</li> <li>10. The maximum number of negative responses to NRC78 for a single diagnostic service is 5.</li> <li>11. Rte mode switching interface of Default/Programming/Extended session.</li> <li>12. Provide the processing flow interface before UDS 10 02 service jumps to Bootloader.</li> <li>13. Not send NRC 0x78 before jumping to Bootloader.</li> <li>14. The maximum number of DIDs that can be read at a time is 5.</li> <li>15. Provide Rte interface for DID support 2E write service.</li> <li>16. Provide Rte interface for DID support IO control 2F service.</li> <li>17. Provide Rte interface for DID support 22 read service.</li> <li>18. Provide Rte interface that DID supports routine control 31 service.</li> <li>19. Provide RTE interface of key seed acquisition and key comparison function for 27 services.</li> <li>20. Provide RTE mode switching interface of ecureset service for ecu reset control.</li> <li>21. Provide RTE mode switching interface of comcontrol service for communication control.</li> <li>22. UpperTest SWC DiagUT for Dcm services</li> </ol>
CanTp	<ol style="list-style-type: none"> <li>1. Auto conf-gen from DEXT.</li> <li>2. Support custom padding, with padding byte "0x00".</li> <li>3. Configure CanTp full duplex mode.</li> <li>4. Configure CanTp addressing mode Normal mode.</li> <li>5. Configure enable the Transmit Cancellation of CanTp.</li> <li>6. Configure the PDU channel</li> <li>7. Physical Request 0x740</li> <li>8. Feature Request 0x7DF</li> <li>9. Physical response 0x748</li> </ol>
Dem	<ol style="list-style-type: none"> <li>1. Auto conf-gen from DEXT.</li> <li>2. Set the operating cycle POWER.</li> <li>3. Provide 1 BSW DTC and 11 ASW DTC examples.</li> <li>4. Snapshot record encoding uses CONFIG type.</li> <li>5. Snapshot record examples.</li> <li>6. Snapshot record data example 0x10B, 0x112, 0xD001, 0xD901, 0xE101.</li> <li>7. Extended data support extended record encoding.</li> <li>8. Extended data support Fault occurrence counter, Fault pending counter, Aged counter and Ageing counter.</li> <li>9. Configuration support NVM storage of DTC.</li> <li>10. DTC storage memory is PrimaryMemory type.</li> <li>11. Provides Rte interface for DTC setting and status reading.</li> <li>12. UpperTest SWC DiagUT for Dem services.</li> </ol>



Functions	Description
NvM	<ol style="list-style-type: none"> <li>1. Auto conf-gen for NvBlock.</li> <li>2. Provides 3 NvBlock examples for application SWC read and write:</li> <li>3. Cyclic write: StoreCyclicBlock</li> <li>4. Immediate write: StoreImmediateBlock</li> <li>5. Shutdown write: StoreAtShutdownBlock</li> <li>6. Support to notify SWC through the callback function after completing a block write operation.</li> <li>7. The NvBlock types are all Native Blocks. Each NvBlock contains 1 RamBlock and 1 RomBlock respectively.</li> <li>8. Support NvM_ReadAll in initialization phase.</li> <li>9. Support to restore default data from ROM when Flash data is damaged during initialization.</li> <li>10. Support shutdown phase NvM_WriteAll.</li> <li>11. Provide test SWC (NvMUT) and test code samples.</li> <li>12. Integration with external Flash Driver</li> </ol>
Measurement Calibration	<ol style="list-style-type: none"> <li>1. Support DBC import with Cobra auto conf-gen.</li> <li>2. Auto deploy XCP PDU to EcuC BSW configuration tool (ISOLAR-B).</li> <li>3. SeedNKey based access authorization for measurement, calibration.</li> <li>4. Timestamp based measurement synchronization.</li> <li>5. Dynamic DAQ.</li> <li>6. Measurement rasters: 10ms,50ms,100ms.</li> <li>7. One-page SERAM Calibration management (WP).</li> <li>8. A2L generation from RTE generated AUTOSAR MCSD.</li> <li>9. UpperTest SWC for measurement and calibration with INCA.</li> </ol>
WdgM	<ol style="list-style-type: none"> <li>1. WdgM alive, deadline, logical supervision with 2 mode: Fastmode and Slowmode.</li> <li>2. Rte mode switch interface to get WdgM global status.</li> <li>3. Rte mode switch interface to get status of each supervised entity.</li> <li>4. Rte interface to get/set WdgM mode.</li> <li>5. Rte interface to get WdgM global supervision status.</li> <li>6. Rte interface to get ID of the supervised entity which first reached the state WDG_M_LOCAL_STATUS_EXPIRED.</li> <li>7. Rte interface to perform WdgM reset.</li> <li>8. Rte interface to get status of each supervised entity.</li> <li>9. WdgIf interface for InternalWatchdog with MCAL integration.</li> <li>10. WdgIf interface for ExternalWatchdog where require user further integration.</li> <li>11. UpperTest SWC for WdgM.</li> </ol>
RTM	<ol style="list-style-type: none"> <li>1. Stack usage monitoring measurement.</li> <li>2. Task profiling monitoring measurement.</li> <li>3. CAT2 ISR profiling monitoring measurement.</li> <li>4. Function profiling monitoring measurement.</li> </ol>
Os*	<ol style="list-style-type: none"> <li>1. SC1 single configuration with multiple OsApplication.</li> <li>2. IoC as cross- OsApplication communication.</li> <li>3. Stack monitor enabled with StackOverrunHook() for user extension.</li> <li>4. Os extended build with ErrorHandler() for user processing of integration errors.</li> <li>5. Os ShutdownHook() for coordination with EcuM_Shutdown().</li> <li>6. DefaultInterruptHandler for any user unhandled interrupts.</li> <li>7. Schedule Table scheduling Rte generated Tasks.</li> </ol>

Functions	Description
Software Partitioning*	<ol style="list-style-type: none"> <li>1. Hierarchical Composition with three compositions: SysCluster, ComCluster, AppCluster</li> <li>2. Software partition with memory map generation.</li> <li>3. Reference linker file and start-up code.</li> </ol>
Build environment	<ol style="list-style-type: none"> <li>1. Python scones build environment.</li> <li>2. Target compilers integration.</li> <li>3. One-click faster build.</li> <li>4. Unique user environment to adapt to other compilers.</li> <li>5. Easy extension for user Hex post treatment.</li> </ol>

Table 3 Feature of BIP

\* Limitations further described in "Limitation" section.

## 2.3 Folder Structure

From table below user could find out which folder is created and shall not be modified manually, and which folders are manual integrated which user shall change during iterative integration.

For <Purpose for user>

- **User shall:** Things that user need to do when use BIP as their project baseline
- **User shall not:** Things that user not do when use BIP as their project baseline
- **User could:** Things that user is recommended to review and adjust on needs

Folder	Subfolder	Description	Purpose for user
ASW	App_SWC	Manual code: Application software components ( <b>empty</b> )	<b>User shall</b> put their ASW source code into this folder
	CalUT	Manual code: UpperTest SWCs for <b>AppCore</b> partition base function: Measurement Calibration	<b>User shall</b> change the variables and parameters implementation on application needs.
	INP_SWC	Manual code: <b>ComCore</b> partition Communication I-Signals reception	<b>User shall</b> change the implementation if I-signal changed from system description.
	OUTP_SWC	Manual code: <b>ComCore</b> partition Communication I-Signals transmission	
	NmUT	Manual code: <b>ComCore</b> partition base function: Network Management	<b>User could</b> modify according to project needs.
	ExeMgrUT	Manual code: <b>SysCore</b> partition base function: Execution Management	<b>User could</b> modify according to project needs.
	DiagUT	Manual code: UpperTest SWCs for <b>SysCore</b> partition base function: Diagnostic	
	NvMUT	Manual code: UpperTest SWCs for <b>SysCore</b> partition base function: Nv Block	
	WdgMUT	Manual code: UpperTest SWCs for <b>SysCore</b> partition base function: Watchdog manager supervision entities.	
BasicSoftware	ISOLAR-AB Project folder for configuration and BSW code generation		

\ecu_config\paramdefs	Arxml configuration: Generated AUTOSAR definition files for BSW.	<b>User shall not</b> modify this folder.
\ecu_config\bsw\static	Arxml configuration: EcuC static configuration files in ISOLAR-B which will be used by RTA-BSW for code generation inputs.	<b>User shall</b> modify this folder on project specific needs.
\ecu_config\bsw\gen	Generated AUTOSAR description and configuration files	<b>User shall not</b> modify this folder.
\ecu_config\bsw\internal	Generated AUTOSAR files	<b>User shall not</b> modify this folder.
\ecu_config\os\static	Arxml configuration: EcuC static configuration files in ISOLAR-B which will be used by RTA-OS for code generation inputs.	<b>User shall</b> modify this folder on project specific needs.
\ecu_config\rte\static	Arxml configuration: EcuC static configuration files in ISOLAR-B which will be used by RTA-RTE for code generation inputs.	<b>User shall</b> modify this folder on project specific needs.
\ecu_config\cdd	Arxml integration: User CDD configuration file folder ( <b>empty</b> )	<b>User shall</b> put their CDD arxml into this folder
\ecu_config\EcuAbstraction	Arxml integration: BIP ECU abstraction layer configuration file folder	<b>User shall</b> modify the ECU abstraction layer arxml on project specific needs.
\swc_config\App_SWC	Application software component description files ( <b>empty</b> )	<b>User shall</b> put their ASW arxml into this folder
\swc_config\CalUT	UpperTest SWCs for <b>AppCore</b> partition base function: Measurement Calibration	<b>User shall</b> change the variables and parameters implementation on application needs.
\swc_config\INP_SWC	Communication I-Signals receive SWC	<b>User shall</b> change the implementation if I-signal changed from system description.
\swc_config\OUTP_SWC	Communication I-Signals transmit SWC	
\swc_config\NmUT	UpperTest SWCs for <b>ComCore</b> partition base function: Network Management	<b>User could</b> modify according to project needs.
\swc_config\DiagUT	UpperTest SWCs for <b>SysCore</b> partition base function: Diagnostic	
\swc_config\ExeMgrUT	SysCore partition base function: Execution Management	
\swc_config\NvBlock	SysCore partition base function: Nv Block	
\swc_config\NvMUT	UpperTest SWCs for SysCore partition base function: Nv Block	
\swc_config\WdgMUT	UpperTest SWCs for SysCore partition base function: Watchdog manager supervision entities.	
\src	Generated BSW code	<b>User shall not</b> modify this folder.
\integration	Integration code either generated by BSW, MCAL or manually created.	<b>User could</b> adapt on project needs (will be describe in Integration Guide <b>section 6 software integration</b> ).

	system_config	Generated system description	<b>User shall not</b> modify this folder.
	<b>DBC</b>	BIP DBC template file	<b>User shall</b> change the templates for PDU and signal description on project needs.
	<b>DEXT</b>	BIP DEXT template file	
CDD	ExtWdg	Manual code: External watchdog wrapper ( <b>stub</b> )	<b>User shall</b> implement these provided CDD drivers stub on their project needs.
	OvrlyM	Manual code: Overlay memory driver ( <b>stub</b> )	
EcuAbstraction	IoHwAbs	IO Hardware abstraction layer	<b>User shall</b> implement these provided IoHwAbs by integrating MCAL standard interfaces on their project needs.
output	Software builds output artefacts.		<b>User shall not</b> modify this folder.
site_scons	Scons build scripts		<b>User shall not</b> modify this folder.
TOOLS	CANoe12	CANoe Test configuration setup	<b>User shall</b> adapt on project needs.
	T32	Example debug script for LAUTERBACH debugger	<b>User shall</b> adapt on project needs.

Table 4 Folder Structure

## 2.4 Targets

The hardware that are used for BIP configuration and integration.

### 2.4.1 Target Hardware

Hardware 1 – verification of generic features	
Manufacturer	Weifu
Board Model Number	N.A.
Target Description	Calterah/CAL60S244AB

Hardware 2 – verification of project specific features (if available, list here)	
Manufacturer	N.A.
Board Model Number	N.A.
Target Description	N.A.

### 2.4.2 Restrictions

No.	Hardware	Description
1	N.A.	N.A.

## 2.5 Tools

The toolchains that are used for BIP configuration and integration.

### 2.5.1 AUTOSAR Configuration Tools

Software	Version	Description
----------	---------	-------------

RTA-CAR	9.2.1	ETAS Classic AUTOSAR solution
ISOLAR-AB	9.2.1	ISOLAR-A: AUTOSAR Authoring tool ISOLAR-B: Basic Software Configuration and code generation tool
RTA-RTE	7.5.3	RTE code generation (ISOLAR-B add-on).
RTA-BSW	6.1.3	BSW module and code generation (ISOLAR-B add-on).
RTA-OS	6.2.0	OS configuration and generation (ISOLAR-B add-on).
RTA-OS ARCMW*	<b>N.A.</b>	<b>RTA-OS port for ARCMW currently not available.</b>
Tresos (3 <sup>rd</sup> party)	28.0.0	Used for MCAL driver configuration and code generation.
MCAL (3 <sup>rd</sup> party)	3.4.0	MCAL from silicon vendor.

Table 5 Configuration Tool

## 2.5.2 Build Tools

Software	Version	Description
Compiler	DesignWare ARC C/C++ Compiler S-2021.12 v11.1 build 003	Compiler for target hardware
Scons (3 <sup>rd</sup> party)	3.0.1	Software builder tool
Python (3 <sup>rd</sup> party)	2.7	Python scripts for running scons build scripts

Table 6 Build Tool

## 2.5.3 Build options

Options updated from last version as highlighted **BLUE** in below table.

Tools	Options
Compiler	-D__MW__ -g -arcv2em -core4 -Hnocopyr -Hnosdata -Hoff=uncached_in_own_section -Hrgf_num_banks=1
Assembler	-Hasopt=-g -arcv2em -core4 -Hnocopyr
Linker	-Os -e_start -core4 -arcv2em -Hldopt=-q -Hldopt=-Cglobals -Hldopt=-Cfunctions -Hldopt=-Csections -Hldopt=-Csize -Hldopt=-Cunmangle

Table 7 Build options

## 2.5.4 RTE Command Line Option

No.	Description
RTE	--strict-unconnected-rport-check=off -nts --os-define-osenv=RTAOS40 --exclusive-area-optimization=enable --os-output-param=changed --notimestamps --use-partition-sections=on -err=xml --deviate-bsw-any-partition=1 --deviate-split-swci-support=on --deviate-prefer-no-empty-executions=on --have-64bit-int-types=on --force-basic-tasks --error-report=xml

Table 8 RTE options

## 2.6 Compatibility

This is the **Master** version from previous version. Except for below differences, other features are compatible with previous version.

No.	Differences	Description
N.A.	N.A.	N.A.

## 2.7 Manual Modifications

The software constraints identified with respect to each release of the project.

No.	File	Solution
1	osNeeds.arxml will generate a OsApplicationMode RTE DEFAULT OS APP by default. However, if this is not needed by user, it can never be disabled.	Delete RTE_DEFAULT_OS_APP from osNeeds.arxml from ISOLAR-B every time Rte generated.

Table 9 Manual Modifications

## 2.8 Fixed Problems

This chapter describes problems fixed with respect to the previous releases of the project.

Issue	Description
N.A.	N.A.

Table 10 Fixed Problems

## 2.9 Limitations

This section describes the set of limitations of each release.

No.	Description
1	<b>OS (Alpha)</b> <ul style="list-style-type: none"> <li>OS is currently provided as source code without configuration tool and OS porting.</li> </ul>
2	<b>Software Partitioning</b> <ul style="list-style-type: none"> <li>Due to linker schema and keywords are currently not fully identified, partitioning with MemMap and linker file is not implemented.</li> </ul>

Table 11 Limitations

## 2.10 Known issues

This section describes the set of known problems of the releases.

Issue ID	Issue description	Impact	Workaround
N.A.	N.A.	N.A.	N.A.

Table 12 Known Issues

## 2.11 Open-Source Software

Please refer to the **OSS** report for details.

## 3 Software Metrics

This software metrics for BIP.

### 3.1.1 Memory Consumption

The total memory footprint of BIP:

- Footprint of RTA-CAR modules (RTA-BSW, RTA-RTE, RTA-OS): **268.32 KB**
- Footprint of MCAL modules: **268.39 KB**

	BIP	MCAL	BSW/OS/RTE
<b>total (KB)</b>	536.8	268.39	268.37
.init_bootstrap/.vector/.text/.rodata	352.8	139.28	213.54
.data/.tls	72.06	26.1	45.96
.bss	111.9	103.01	8.87

### 3.1.2 CPU Load

The CPU runtime	
CPU Frequency	300 MHz
CPU Utilization (Load)	22 %

### 3.1.3 Stack Consumption

The stack usage (%)			
ARC EM6 single core	SYS_STACK (4KB)	OS Category 1 ISR stacks	10%
		OS Category 2 ISR stacks	
		AUTOSAR Applications	

## 4 ETAS Contact Addresses

---

### *ETAS HQ*

---

ETAS GmbH

Borsigstraße 14

70469 Stuttgart

Germany

Phone: +49 711 3423-0

Fax: +49 711 3423-2106

WWW: [www.etas.com](http://www.etas.com)

### *ETAS Automotive Technology (Shanghai) Co., Ltd.*

---

333 Fuquan Road North, IBP

Changning District

Shanghai

China

Phone: +86 21 2218 5858

Fax: +86 21 5037 2221

WWW: [www.etas.com](http://www.etas.com)



