



# **Omc Classic Integration Manual**

Project BMW AUTOSAR 4 Core Rel. 3

Author BMW AG

Release Date 2017-12-14

version 5.2.0

Status Release

Hotline +49 89 382 - 32233

contact bac@bmw.de

https://asc.bmw.com/jira/browse/BSUP (extern) https://asc.bmwgroup.net/jira/browse/BSUP (intern)

# **Revision History**

Version	Date	Description
5.2.0	2017-12-14	BAC-6565: document callback mechanism to establish intrinsic safety
		(AllowModeChange and OmcOperatingModeCallout)
5.1.1	2017-10-12	Version Update
5.1.0	2017-08-10	Version Update
5.0.0	2017-06-29	Initial version for SP2021

**Company** Bayerische Motoren Werke Aktiengesellschaft

Postal address BMW AG 80788 München

Office address
Forschungs- und
Innovationszentrum

(FIZ) Hufelandstr. 1 80937 München

**Telephone** Switchboard +49 89 382-0

Internet www.bmwgroup.com



# **Table of Contents**

1	Introduction	4
	1.1 Functional overview	4
2	Related documentation	5
3	Limitations	6
4	Software Architecture	7
	4.1 Dependencies on AUTOSAR modules	7
	4.1.1 RTE	7
	4.1.2 Det	7
	4.1.3 Dcm	7
	4.1.4 Dem	7
	4.1.5 Nvm	7
	4.2 Dependencies to other modules	7
5	Integration	8
	5.1 Configuration of other Modules	8
	5.1.1 Dcm	
	5.1.1.1 Read Data By Identifer	8
	5.1.1.2 Routine Control	8
	5.1.2 NvM	8
	5.1.3 Dem	9
	5.1.3.1 Enable Condition	9
	5.1.3.2 Event	
	5.1.4 BswM	
	5.1.5 Det	
	5.2 Provided Interfaces	
	5.2.1 OmcOperatingMode and OmcExtendedOperatingMode	
	5.2.2 OmcModesCalloutsResult	
	5.3 Required Interfaces	
	5.3.1 OmcModesCallouts	
	5.4 Configuration	
	5.4.1 OmcGeneral	
	5.4.1.1 OmcDevErrorDetect	
	5.4.1.2 OmcOperatingModeCallout	
	5.5 Configuration of the RTE	
	5.5.1 Event Mapping	
	5.5.2 Data Mapping	
	5.5.2.1 Dcm	
	5.5.2.2 Det	
	5.5.3 Dem	
	5.5.4 NvM	
	5.5.5 BswM	
	5.5.6 Exclusive Areas	
	5.6 Software Integration	
	5.6.1 Startup/Initialization	12





5.6.2	Normal Operation	12
5.6.3	Shutdown/Deactivation	12





# 1 Introduction

This Integration Manual describes the basis functionality, API and the configuration and integration of the BMW System Function Omc.

### **Functional overview**

The main objective of the Omc functionality is to maintain the current Operating Mode of an ECU.

This means:

- Allowing the change of the current Operating Mode via diagnostic request
- Saving the current Operating Mode to Non Volatile RAM (NVRAM)
- Enabling/disabling Dem (more concrete: Setting/Unsetting a enable condition)
- Providing the current Operating Mode to other software components.

The Omc module distinguishes the following vehicle operating modes:

- NORMAL
- ASSEMBLY
- TRANSPORT
- FLASH

Note that in different documents these modes are sometimes called "energy modes" sometimes called "operating modes". Although both terms are equivalent, we strictly use the term operating mode.



# 2 Related documentation

# References



# 3 Limitations

No limitations are known.





# 4 Software Architecture

# **Dependencies on AUTOSAR modules**

The current version of the Module Omc depends on the following BSW modules:

### **RTE**

As a software component, the Omc module uses Rte client/server communication to communicate with other SWCs and BSW. Additionally the scheduling is done by the Rte.

#### Det

In case Det usage is enabled in the Omc configuration, Omc will report development errors by using the Det functionality.

#### **Dcm**

The Dcm will call functionality of the Omc module when a RDBI for the operating mode or the extended operating mode has been received. The corresponding R-ports of the Dcm for these two identifiers must be connected with the corresponding P-ports of the Omc.

### Dem

The Omc use the operation SetEventStatus of the CSI DiagnosticMonitor to set an Event via Rte. It also uses the ClientServer Interface of service EnableCondition.

#### Nvm

The Omc relies on services provided by the Nvm to store its persistent data regarding the current operating mode and current extended operating mode.

### **Dependencies to other modules**

Omc does not have dependencies to other modules.



# 5 Integration

# **Configuration of other Modules**

The following modules shall be configured, before this module can be generated, compiled and linked.

#### **Dcm**

## **Read Data By Identifer**

[] [Two Read Data By Identifier commands shall be configured within Dcm (22 10 0A and 22 10 0E). |(DK\_T3\_736, DK\_T3\_762, FZM\_SC\_SYS\_PA\_335, DK\_T3\_1255)

- DcmDspDataSize shall be set to 8
- DcmDspDataUsePort shall be set to USE\_DATA\_SYNCH\_CLIENT\_SERVER
- DID shall be configured to be read in all sessions and security levels

#### **Routine Control**

[] [Two Routine Control commands shall be configured within Dcm (31 01 0F 0C and 31 01 10 03). ](DK\_T3\_720, DK\_T3\_725, DK\_T3\_727, DK\_T3\_729, DK\_T3\_751, FZM\_SC\_SYS\_PA\_334, FZM\_SC\_SYS\_PA\_334)

- DcmDspRoutineFixedLength shall be true
- DcmDspRoutineUsePort shall be true
- Only DcmDspStartRoutineIn shall be configured
  - DcmDspRoutineSignalLength shall be set to 8
  - DcmDspRoutineSignalPos shall be set to 0

#### NvM

The Omc needs a Nvm block to store the operating mode:

# 

- block size shall be set to 2 bytes
- Ram block address shall be set to Omc\_NvData
- Rom block address shall be set to Omc\_DefaultNvData
- NvmBlockManagementType shall be set to NVM\_BLOCK\_NATIVE
- NvmBlockUseCrc shall be disabled
- NvmSelectBlockForReadall shall be set to true

J(DK\_T3\_743, FZM\_SC\_SYS\_PA\_169, FZM\_SC\_SYS\_PA\_329, FZM\_SC\_SYS\_PA\_171, FZM\_SC\_SYS\_PA\_331)



#### Dem

#### **Enable Condition**

A EnableCondition for Omc shall be configured wihtin Dem.

This EnableCondition shall be referenced by all EnableConditionGroups in the system, that depend on this EnableCondition.

Example: At BMW the permission to report communication error relevant Dem events depends on the state of different sources (operating mode, diagnostic state and Stm centralErrorLock). Each of these sources is connected to its own EnableCondition and all these conditions are combined in one EnableConditionGroup.

At the end, all Dem Events that represent communication errors shall reference this EnableGroup via their DemEnableConditionGroupRef parameter.

#### **Event**

An Dem event shall be configured for Omc.

### 

- DTC vale shall be set to 0x020000 + 0x100 \* ECU\_Address
- Event destination: primary origin
- Event kind: DEM\_EVENT\_KIND\_SWC
- A Clear Event Allowed port shall be configured via Rte
- Operation Cycle: this event shall reference an operation cycle which is started before OmcMode is INIT and stays in this state until OmcMode is STOP.

(DK\_T3\_711)

#### **BswM**

The BswM controls the states of the Omc module.

- BswMModeRequestPort for the Omc operation mode
- BswMRules to switch the Omc operation mode
- BswMRteSwitch actions for the Omc operation mode

## Det

A Omc entry shall be added to the Software Component List from Det.





# **Provided Interfaces**

## OmcOperatingMode and OmcExtendedOperatingMode

The Omc provides two RTE Modes which indicate the operating and extended operating mode which are currently active.

#### **OmcModesCalloutsResult**

This interface may be used by the user if OmcOperatingModeCallout is set to true when the interfaces AllowOpModeSwitch or OpModeSwitchCancelled returns RUNNING.

The user may indicate over this interface using the following operations:

 $- \ \, OpModeSwitchAllowedResult: \ \, shall \ \, be \ \, called \ \, if \ \, AllowOpModeSwitch \ \, returned \\ E_RUNNINGtoinformtheOmcifamodeorextendedmodeswitchisallowedornot. OpModeSwitchCancelledAck \\ shallbecalledifOpModeSwitchCancelledreturnedE_RUNNINGtoinformtheOmcthatthemodeorextendedmodeswitchCancelledAck \\ shallbecalledifOpModeSwitchCancelledreturnedE_RUNNINGtoinformtheOmcthatthemodeorextendedmodeswitchisallowedornot. \\$ 

## **Required Interfaces**

#### **OmcModesCallouts**

The user has to provide this this interface if the parameter OmcOperatingModeCallout is set to true.

The following two opereations shall be implemented:

- AllowOpModeSwitch: is called by Omc during a diagnose request to change the operating or extended operating mode. The user will be able to accept or deny this request.
- OpModeSwitchCancelled: if the interface AllowOpModeSwitch has been called and the operating or extended operating mode change is not performed the user is informed over this interface that the change has been cancelled. This may happens e.g. if an error occurs while trying to store the new operating mode in non volatile memory.

# Configuration

### **OmcGeneral**

This container contains the configuration parameters of the Omc

#### **OmcDevErrorDetect**

Activate/Deactivate the Development Error Detection and Notification.

If set to true: Development Error Detection and Notification activated. If set to false: Development Error Detection and Notification deactivated.





## **OmcOperatingModeCallout**

Activate/Deactivate an user callback to be able to accept or deny a mode or extended mode change. If set to true: The application provides one or more instances of the service AllowModeChange. The user may control over this interface if mode or extended mode change is allowed or not. The user is only called to allow or not a mode or extended mode change only if the mode change has already be accepted by StdDiag. If set to false: Omc decides with StdDiag if a mode or extended mode change allowed is or not.

# **Configuration of the RTE**

## **Event Mapping**

The followings runnable entities shall be mapped:

- Init
- RollbackModeChange

# **Data Mapping**

#### **Dcm**

- The port ControlEnergySavingState shall be connected to the Dcm generated RoutineServices\_<xxx> see 6.1 Dcm, Routine Control.
- The port ControlExtendedEnergySavingState shall be connected to the Dcm generated RoutineServices <xxx> see 6.1 Dcm, Routine Control.
- The port EnergySavingState shall be connected to the Dcm generated DataServices\_<xxx> see 6.1
   Dcm, Read Data By Identifier
- The port ExtendedEnergySavingState shall be connected to the Dcm generated DataServices\_<xxx> see 6.1 Dcm, Read Data By Identifier.

#### Det

- The port DETServicePort shall be connected to the Det generated service port see 6.1 Dem, Det.

#### Dem

- The port ClearEventAllowedPort shall be connected to the Dem generated clear event allowed port see 6.1 Dem, Event.
- The port operatingModeEvent shall be connected to the Dem generated Diagnostic Monitor port see 6.1 Dem, Event.

#### NvM

The port NvMServicePort shall be connected to the NvM generated service port see 6.1. NvM.

#### **BswM**

- The port notificationOMCModePort shall be connected to the BswM generated mode switch port see 6.1 BswM.
- The prot requestOmcModePort shall be connected to the BswM generated mode request port see
   6.1. BswM.





#### **Exclusive Areas**

Omc uses one Exclusive Area called Mode.

# **Software Integration**

## Startup/Initialization

The NvM ReadAll shall be complete before the Omc mode OMC\_INITIALIZED is requested. The initialization is complete when the mode switch to OMC\_INITIALIZED (triggered by Omc) is completed. While in OMC\_INITIALIZED the integrator shall decide the best moment to BswM to request OMC\_RUNNING. After requesting OMC\_RUNNING Omc will switch the mode to OMC\_RUNNING. When the switch is completed the Omc will be running.

To request a specific state the user shall use the port LifeCycleRequest.

### **Normal Operation**

#### Shutdown/Deactivation

The BswM shall request the Omc Mode to OMC\_STOPPED to stop the Omc SW-C. The module is stopped when the mode switch to OMC\_STOPPED (triggered by Omc) is completed.