

| Document Title             | Requirements on OCU Driver |
|----------------------------|----------------------------|
| Document Owner             | AUTOSAR                    |
| Document Responsibility    | AUTOSAR                    |
| Document Identification No | 614                        |
| Document Classification    | Auxiliary                  |
| Document Version           | 1.0.1                      |
| Document Status            | Final                      |
| Part of Release            | 4.1                        |
| Revision                   | 2                          |

| Document Change History |         |                                  |   |
|-------------------------|---------|----------------------------------|---|
| Date                    | Version | Changed by                       | Change Description  |
| 31.10.2013              | 1.0.1   | AUTOSAR<br>Release<br>Management | <ul style="list-style-type: none"><li>Editorial changes</li></ul> |
| 18.02.2013              | 1.0.0   | AUTOSAR<br>Administration        | <ul style="list-style-type: none"><li>Initial Release</li></ul>   |

## **Disclaimer**

This specification and the material contained in it, as released by AUTOSAR, is for the purpose of information only. AUTOSAR and the companies that have contributed to it shall not be liable for any use of the specification.

The material contained in this specification is protected by copyright and other types of Intellectual Property Rights. The commercial exploitation of the material contained in this specification requires a license to such Intellectual Property Rights.

This specification may be utilized or reproduced without any modification, in any form or by any means, for informational purposes only.

For any other purpose, no part of the specification may be utilized or reproduced, in any form or by any means, without permission in writing from the publisher.

The AUTOSAR specifications have been developed for automotive applications only. They have neither been developed, nor tested for non-automotive applications.

The word AUTOSAR and the AUTOSAR logo are registered trademarks.

## **Advice for users**

AUTOSAR specifications may contain exemplary items (exemplary reference models, "use cases", and/or references to exemplary technical solutions, devices, processes or software).

Any such exemplary items are contained in the specifications for illustration purposes only, and they themselves are not part of the AUTOSAR Standard. Neither their presence in such specifications, nor any later documentation of AUTOSAR conformance of products actually implementing such exemplary items, imply that intellectual property rights covering such exemplary items are licensed under the same rules as applicable to the AUTOSAR Standard.

## Table of Contents

|           |  |           |
|-----------|--|-----------|
| <b>1</b>  | <b>Scope of this document</b>  | <b>5</b>  |
| <b>2</b>  | <b>How to read this document</b>   | <b>6</b>  |
| 2.1       | Conventions used   | 6         |
| 2.2       | Requirements structure   | 7         |
| <b>3</b>  | <b>Acronyms and abbreviations</b>  | <b>8</b>  |
| <b>4</b>  | <b>Requirement Specification</b>   | <b>9</b>  |
| 4.1       | OCU Driver   | 9         |
| 4.1.1     | Functional Overview  | 9         |
| 4.1.2     | Functional Requirements  | 10        |
| 4.1.2.1   | Configuration  | 10        |
| 4.1.2.1.1 | [SRS_Ocu_00002] The OCU driver shall support the following basic static configurations per channel   | 10        |
| 4.1.2.2   | Initialization   | 10        |
| 4.1.2.2.1 | [SRS_Ocu_00004] After initialization of the OCU driver, all notifications shall be disabled  | 10        |
| 4.1.2.2.2 | [SRS_Ocu_00005] The OCU Driver shall provide the functionality to de-initialize OCU driver   | 11        |
| 4.1.2.3   | Normal Operation   | 11        |
| 4.1.2.3.1 | [SRS_Ocu_00006] The OCU driver shall provide a notification for an OCU channel when the current value of the counter matches the threshold             | 11        |
| 4.1.2.3.2 | [SRS_Ocu_00007] The OCU driver shall allow enabling /disabling notifications for an OCU channel during runtime   | 12        |
| 4.1.2.3.3 | [SRS_Ocu_00008] The OCU driver shall provide services for starting and stopping a channel  | 12        |
| 4.1.2.3.4 | [SRS_Ocu_00009] The OCU driver shall provide a synchronous service for reading the value of the counter  | 13        |
| 4.1.2.3.5 | [SRS_Ocu_00010] The OCU driver shall provide services to modify the value of the threshold of a channel  | 13        |
| 4.1.2.3.6 | [SRS_Ocu_00011] The OCU driver shall provide a synchronous service to set the state of the output pin attached to a channel                            | 13        |
| 4.1.2.3.7 | [SRS_Ocu_00012] The OCU driver shall provide a service to set the action that will be performed by the pin attached to a channel upon comparison match | 14        |
| 4.1.2.3.8 | [SRS_Ocu_00013] The OCU Driver shall allow configuring the same counter base and timing to be used by several channels                                 | 14        |
| 4.1.2.3.9 | [SRS_Ocu_00014] All units used within the API services of the OCU driver shall be of unit ticks  | 14        |
| 4.1.2.4   | Shutdown Operation   | 15        |
| <b>5</b>  | <b>Related Documentation</b>   | <b>16</b> |
| 5.1       | Autosar deliverables   | 16        |
| 5.2       | Related standards and norms  | 16        |
| 5.2.1     | HIS  | 16        |



## 1 Scope of this document

This document specifies requirements on the module **OCU Driver**.

### Constraints

First scope for specification of requirements on basic software modules is systems which are not safety relevant. For this reason safety requirements are assigned to medium priority.

## 2 How to read this document

Each requirement has its unique identifier starting with the prefix “BSW” (for “Basic Software”). For any review annotations, remarks or questions please refer to this unique ID rather than chapter or page numbers!

### 2.1 Conventions used

- The representation of requirements in AUTOSAR documents follows the table specified in [6].
- In requirements, the following specific semantics are used

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted . Note that the requirement level of the document in which they are used modifies the force of these words.

- **MUST:** This word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification.
- **MUST NOT:** This phrase, or the phrase „SHALL NOT“, means that the definition is an absolute prohibition of the specification.
- **SHOULD:** This word, or the adjective "RECOMMENDED", mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- **SHOULD NOT:** This phrase, or the phrase "NOT RECOMMENDED" mean that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
- **MAY:** This word, or the adjective „OPTIONAL“, means that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation, which does not include a particular option, **MUST** be prepared to interoperate with another implementation, which does include the option, though perhaps with reduced functionality. In the same vein an implementation, which does include a particular option, **MUST** be prepared to interoperate with another implementation, which does not include the option (except, of course, for the feature the option provides.)

## 2.2 Requirements structure

Each module specific chapter contains a short functional description of the Basic Software Module. Requirements of the same kind within each chapter are grouped under the following headlines (where applicable):

Functional Requirements:

- Configuration (which elements of the module need to be configurable)
- Initialization
- Normal Operation
- Shutdown Operation
- Fault Operation
- ...

Non-Functional Requirements:

- Timing Requirements
- Resource Usage
- Usability
- Output for other WPs (e.g. Description Templates, Tooling...)
- ...

### 3 Acronyms and abbreviations

Acronyms and abbreviations that have a local scope are not contained in the AUTOSAR glossary. These must appear in a local glossary.

| <b>Acronym</b>       | <b>Description</b>   |
|----------------------|--|
| OCU                  | Output Compare Unit  |
| OCU channel          | Represents a logical entity composed of a free running counter a comparison threshold and the action that is done as a result of the comparison process. |
| Compare threshold.   | Target value that is compared with the content of the counter each time the counter increases by one unit.   |
| Free running counter | A counter that runs from a minimum to a maximum value and restarts automatically to the minimum after reaching the maximum value.                        |
| DMA                  | Direct memory access   |
| HIS                  | Herstellerinitiative Software  |
| MCAL                 | Microcontroller Abstraction Layer  |
| MCU                  | Microcontroller Unit   |
| SPAL                 | Standard Peripheral Abstraction Layer  |

| <b>Abbreviation</b> | <b>Description</b>                |
|---------------------|-----------------------------------|
| STD                 | Standard                          |
| UNINIT              | Uninitialized (= not initialized) |

As this is a document from professionals for professionals, all other terms are expected to be known.



## 4 Requirement Specification

### 4.1 OCU Driver

#### 4.1.1 Functional Overview

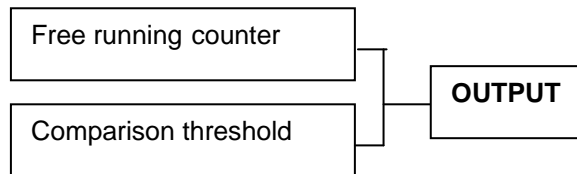
The driver provides a means to compare two values and act automatically upon compare match, allowing therefore speeding up software processes.

This specification specifies the functionality, API and the configuration of the AUTOSAR Basic Software module Output Compare driver (OCU).

Each OCU channel is linked to hardware OCU which belongs to the microcontroller.

The driver provides services for initialization and control of the microcontroller internal OCU peripheral. The OCU module can generate HW signals to drive external devices.

The picture below shows the typical representation of an OCU channel.



**Figure 1: Output Compare channel**

The 'output' is the action that is actually done upon compare match. A software action and/or a hardware action can be performed according to the configuration.

## 4.1.2 Functional Requirements

### 4.1.2.1 Configuration

#### 4.1.2.1.1 [SRS\_Ocu\_00002] The OCU driver shall support the following basic static configurations per channel

|                             |   |
|-----------------------------|---|
| <b>Type:</b>                | Valid   |
| <b>Description:</b>         | <p>The OCU driver shall support the following basic static configurations per channel:</p> <p>Mandatory parameters</p> <ul style="list-style-type: none"> <li>• Symbolic name for channel / channel ID.</li> <li>• Maximum value of the counter</li> <li>• count resolution/frequency</li> <li>• Notification function(s)</li> <li>• Default value of the threshold</li> <li>• Assigned hardware channel.</li> </ul> <p>Optional parameters</p> <ul style="list-style-type: none"> <li>• Count direction</li> <li>• Output signal (internal signal or port pin, if provided by hardware).<br/>The default output level (value after reset) shall be available. <ul style="list-style-type: none"> <li>• OCU_LOW/OCU_HIGH,</li> </ul> </li> <li>• Hardware event(s) triggered by the channel (if supported by HW). <ul style="list-style-type: none"> <li>• The hardware resource ID: (e.g OCU_ADC, OCU_DMA)</li> <li>• For each hardware resource the adequate number.<br/>e.g: ADC_chn1.</li> </ul> </li> <li>• Optional clock settings if supported by hardware.</li> </ul> <p>In addition, the OCU driver shall allow the configuration OCU-peripheral specific settings if supported by hardware.</p> |
| <b>Rationale:</b>           | To allow different usage for each channel   |
| <b>Use Case:</b>            |   |
| <b>Dependencies:</b>        | --  |
| <b>Supporting Material:</b> |   |

] (RS\_BRF\_01888,RS\_BRF\_01904,RS\_BRF\_01136)

### 4.1.2.2 Initialization

#### 4.1.2.2.1 [SRS\_Ocu\_00004] After initialization of the OCU driver, all notifications shall be disabled

|                             |   |
|-----------------------------|---|
| <b>Type:</b>                | Valid   |
| <b>Description:</b>         | <p>After initialization of the OCU driver, all notifications shall be disabled. All channels shall be stopped. (No counting running).<br/>In case a channel has an associated output pin, this pin shall be set to the default value defined in the configuration of the channel.</p> |
| <b>Rationale:</b>           | To prevent unwanted events after initialization.  |
| <b>Use Case:</b>            | --  |
| <b>Dependencies:</b>        | --  |
| <b>Supporting Material:</b> | --  |

] (RS\_BRF\_01096)

#### 4.1.2.2.2 [SRS\_Ocu\_00005] The OCU Driver shall provide the functionality to de-initialize OCU driver

|                             |   |
|-----------------------------|---|
| <b>Type:</b>                | Valid   |
| <b>Description:</b>         | The OCU Driver shall provide the functionality to de-initialize OCU driver.   |
| <b>Rationale:</b>           | It is necessary to reset the driver specific resources (SFRs, variables ...) before a valid initialization can be done. |
| <b>Use Case:</b>            | De-initialization is needed before re-initialization with a valid configuration (post-build) during ECU runtime.        |
| <b>Dependencies:</b>        | --  |
| <b>Supporting Material:</b> | --  |

] (RS\_BRF\_01096, RS\_BRF\_01056)

#### 4.1.2.3 Normal Operation

##### 4.1.2.3.1 [SRS\_Ocu\_00006] The OCU driver shall provide a notification for an OCU channel when the current value of the counter matches the threshold

|                             |  |
|-----------------------------|--|
| <b>Type:</b>                | Valid  |
| <b>Description:</b>         | The OCU driver shall provide a notification for an OCU channel when the current value of the counter matches the threshold. The notification should be triggered under the following conditions: <ul style="list-style-type: none"> <li>• The notification function configured as non null pointer</li> <li>• and only if notifications are enabled</li> </ul> |
| <b>Rationale:</b>           | Upon comparison matches on a channel, an action must be performed. The module that will perform the action needs to be informed.   |
| <b>Use Case:</b>            | Upper layers perform actions synchronized to a reference, which is represented by the current value of the threshold   |
| <b>Dependencies:</b>        | --   |
| <b>Supporting Material:</b> |  |

] (RS\_BRF\_01064, RS\_BRF\_01968, RS\_BRF\_01056)

#### 4.1.2.3.2 [SRS\_Ocu\_00007] The OCU driver shall allow enabling /disabling notifications for an OCU channel during runtime

[

|                             |   |
|-----------------------------|---|
| <b>Type:</b>                | Valid   |
| <b>Description:</b>         | The OCU driver shall allow enabling /disabling notifications for an OCU channel during runtime.<br>The following options shall be available for each selected channel: <ul style="list-style-type: none"> <li>• disable notification</li> <li>• enable notification on compare match (current value of the counter equals threshold value)</li> </ul> |
| <b>Rationale:</b>           | Prevents from invoking unwanted notifications (interrupts) and allows selecting the exact point in the execution flow of the software when the first or next notification may arise.  |
| <b>Use Case:</b>            | When the channel counter matches the defined threshold, the upper layer is informed to perform a task and can change the current value of the threshold   |
| <b>Dependencies:</b>        | --  |
| <b>Supporting Material:</b> |   |

] (RS\_BRF\_01056, RS\_BRF\_01152)

#### 4.1.2.3.3 [SRS\_Ocu\_00008] The OCU driver shall provide services for starting and stopping a channel

[

|                             |   |
|-----------------------------|---|
| <b>Type:</b>                | Valid   |
| <b>Description:</b>         | The OCU driver shall provide services for starting and stopping a channel. Stopping the channel means no more actions shall be triggered. Starting a channel means allowing the configured actions to be performed. |
| <b>Rationale:</b>           | Allow for software to control operations individually on each channel.  |
| <b>Use Case:</b>            | An H-bridge is connected to the pin of a channel. The pin must start driving the bridge only when this one is ready to operate.   |
| <b>Dependencies:</b>        | --  |
| <b>Supporting Material:</b> |   |

] (RS\_BRF\_01056, RS\_BRF\_01096, RS\_BRF\_01136)

#### 4.1.2.3.4 [SRS\_Ocu\_00009] The OCU driver shall provide a synchronous service for reading the value of the counter

|                             |   |
|-----------------------------|---|
| <b>Type:</b>                | Valid   |
| <b>Description:</b>         | The OCU driver shall provide a synchronous service for reading the value of the counter.  |
| <b>Rationale:</b>           | The counter value of a channel corresponds to physical information (time, pressure, angular position...) which is used by the upper layer for various purposes. |
| <b>Use Case:</b>            | The upper layer reads the current value of the counter and computes the corresponding physical value.   |
| <b>Dependencies:</b>        | --  |
| <b>Supporting Material:</b> |   |

] (RS\_BRF\_01056)

#### 4.1.2.3.5 [SRS\_Ocu\_00010] The OCU driver shall provide services to modify the value of the threshold of a channel

|                             |   |
|-----------------------------|---|
| <b>Type:</b>                | Valid   |
| <b>Description:</b>         | The OCU driver shall provide services to modify the value of the threshold of a channel. This service shall allow to write for the threshold: <ul style="list-style-type: none"> <li>• an absolute value</li> <li>• or a relative value: with respect to the current counter value</li> <li>• or a relative value: with respect to the current threshold value.</li> </ul> Relative value setting needs hardware support. |
| <b>Rationale:</b>           | Provide the upper layer with a means to change dynamically the comparison threshold of a channel to adapt to the running software.  |
| <b>Use Case:</b>            | Update the next comparison threshold with a relative value when the counter rolls over.   |
| <b>Dependencies:</b>        | --  |
| <b>Supporting Material:</b> | --  |

] (RS\_BRF\_01056,RS\_BRF\_01152)

#### 4.1.2.3.6 [SRS\_Ocu\_00011] The OCU driver shall provide a synchronous service to set the state of the output pin attached to a channel

|                             |  |
|-----------------------------|--|
| <b>Type:</b>                | Valid  |
| <b>Description:</b>         | The OCU driver shall provide a synchronous service to set the state of the output pin attached to a channel.       |
| <b>Rationale:</b>           | According to different phases of a system (SHUTDOWN, IDLE...) the output pin must have a known state (LOW or HIGH) |
| <b>Use Case:</b>            | Initialize the pin associated to a channel to the relevant level.  |
| <b>Dependencies:</b>        | --   |
| <b>Supporting Material:</b> |  |

] (RS\_BRF\_01056,RS\_BRF\_01152,RS\_BRF\_01096,RS\_BRF\_01888)

#### 4.1.2.3.7 [SRS\_Ocu\_00012] The OCU driver shall provide a service to set the action that will be performed by the pin attached to a channel upon comparison match

|                             |  |
|-----------------------------|--|
| <b>Type:</b>                | Valid  |
| <b>Description:</b>         | The OCU driver shall provide a service to preset the action that will be performed by the pin attached to a channel upon compare match if supported by HW. This service shall take as parameters: <ul style="list-style-type: none"> <li>• OCU channel</li> <li>• Pin action upon comparison (if supported by HW).</li> </ul> Possible values of the pin action shall be OCU_HIGH, OCU_LOW, OCU_TOGGLE, OCU_DISABLE. |
| <b>Rationale:</b>           | The output pin of a channel must behave in accordance with the system needs.   |
| <b>Use Case:</b>            | Generate pulses at the output pin of a channel.  |
| <b>Dependencies:</b>        | --   |
| <b>Supporting Material:</b> | --   |

] (RS\_BRF\_01056,RS\_BRF\_01152,RS\_BRF\_01888)

#### 4.1.2.3.8 [SRS\_Ocu\_00013] The OCU Driver shall allow configuring the same counter base and timing to be used by several channels

|                             |   |
|-----------------------------|---|
| <b>Type:</b>                | Valid   |
| <b>Description:</b>         | The OCU Driver shall allow configuring the same counter base and timing to be used by several channels, if this functionality is supported by hardware. This will allow using several channels in a group. The configuration shall provide the following parameters <ul style="list-style-type: none"> <li>• OCU channels to group</li> </ul> |
| <b>Rationale:</b>           | Allow synchronizing different actions which have the same reference.  |
| <b>Use Case:</b>            | Drive synchronously a pair of switches that are part of an H-bridge.  |
| <b>Dependencies:</b>        | --  |
| <b>Supporting Material:</b> | --  |

] (RS\_BRF\_01632, RS\_BRF\_01904)

#### 4.1.2.3.9 [SRS\_Ocu\_00014] All units used within the API services of the OCU driver shall be of unit ticks

|                             |   |
|-----------------------------|---|
| <b>Type:</b>                | Valid   |
| <b>Description:</b>         | All units used within the API services of the OCU driver shall be of unit ticks.  |
| <b>Rationale:</b>           | Conversions between the physical units (of the threshold and counter) and ticks shall be part of the ECU Abstraction Layer. |
| <b>Use Case:</b>            | --  |
| <b>Dependencies:</b>        | --  |
| <b>Supporting Material:</b> |   |

] (RS\_BRF\_02040)

#### **4.1.2.4 Shutdown Operation**

None.

## **5 Related Documentation**

### **5.1 Autosar deliverables**

- [1]** Glossary  
AUTOSAR\_TR\_Glossary.pdf
- [2]** Layered Software Architecture  
AUTOSAR\_EXP\_LayeredSoftwareArchitecture.pdf
- [3]** General Requirements on Basic Software  
AUTOSAR\_SRS\_BSWGeneral.pdf
- [4]** General Requirements on SPAL  
AUTOSAR\_SRS\_SPALGeneral.pdf
- [5]** List of Basic software modules  
AUTOSAR\_BasicSoftwareModuleList.pdf
- [6]** Software Standardization Template  
AUTOSAR\_TPS\_StandardizationTemplate.pdf

### **5.2 Related standards and norms**

#### **5.2.1 HIS**

- [7]** HIS API I/O Driver Specification V2.1.3