

---

---

**Road vehicles — Data communication  
between sensors and data fusion unit  
for automated driving functions —  
Logical interface**

*Véhicules routiers — Communication de données entre capteurs et  
unité de fusion de données pour les fonctions de conduite automatisée  
— Interface logique*

STANDARDSISO.COM : Click to view the full PDF of ISO 23150:2023



STANDARDSISO.COM : Click to view the full PDF of ISO 23150:2023



**COPYRIGHT PROTECTED DOCUMENT**

© ISO 2023

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office  
CP 401 • Ch. de Blandonnet 8  
CH-1214 Vernier, Geneva  
Phone: +41 22 749 01 11  
Email: [copyright@iso.org](mailto:copyright@iso.org)  
Website: [www.iso.org](http://www.iso.org)

Published in Switzerland

# Contents

Page

|  |           |
|--|-----------|
| Foreword.....  | vi        |
| Introduction.....  | viii      |
| <b>1 Scope .....</b>   | <b>1</b>  |
| <b>2 Normative references .....</b>  | <b>1</b>  |
| <b>3 Terms and definitions.....</b>  | <b>1</b>  |
| 3.1 Architectural components.....  | 1         |
| 3.2 Terms for logical interface layers.....  | 2         |
| 3.3 Structure terms .....  | 3         |
| 3.4 Measurement terms.....   | 3         |
| 3.5 Requirement level terms .....  | 6         |
| 3.6 Road user relevant entity types .....  | 7         |
| 3.7 Axis and coordinate system terms .....   | 9         |
| <b>4 Abbreviated terms .....</b>   | <b>15</b> |
| <b>5 Structure of the interface description.....</b>                                       | <b>16</b> |
| 5.1 General .....  | 16        |
| 5.2 Signal .....   | 17        |
| 5.3 Interface.....   | 17        |
| 5.4 Specific signal grouping.....  | 18        |
| 5.5 Profile .....  | 19        |
| <b>6 Logical interface from a sensor as well as a sensor cluster to a fusion unit.....</b> | <b>19</b> |
| 6.1 General .....  | 19        |
| 6.2 Generic interface header .....   | 22        |
| 6.3 Generic interface entity.....  | 22        |
| 6.4 Profile: Uniqueness of interface versioning.....                                       | 23        |
| <b>7 Object level.....</b>   | <b>23</b> |
| 7.1 General .....  | 23        |
| 7.2 Generic object level interface.....  | 24        |
| 7.2.1 Generic object level header .....  | 24        |
| 7.2.2 Generic object level entity .....  | 25        |
| 7.3 Potentially moving object interface.....   | 25        |
| 7.3.1 Potentially moving object header.....  | 34        |
| 7.3.2 Potentially moving object entity .....   | 35        |
| 7.3.3 Profile: Motion type .....   | 36        |
| 7.3.4 Profile: Motion state vector .....   | 37        |
| 7.4 Road object interface .....  | 38        |
| 7.4.1 Road object header .....   | 49        |
| 7.4.2 Road object entity.....  | 51        |
| 7.4.3 Profile: Colour model for RDOI.....  | 53        |
| 7.5 Static object interface.....   | 53        |
| 7.5.1 Static object header .....   | 81        |
| 7.5.2 Static object entity.....  | 83        |
| 7.5.3 Profile: Colour model for SOI.....   | 91        |
| 7.5.4 Profile: Detection references for 3D detections .....                                | 91        |
| 7.6 Free space area object interface .....   | 94        |
| 7.6.1 Free space area object header .....  | 99        |
| 7.6.2 Free space area object entity.....   | 100       |

|        |  |     |
|--------|--|-----|
| 8      | Feature level .....                                      | 101 |
| 8.1    | General.....   | 101 |
| 8.2    | Generic sensor cluster feature interface .....           | 102 |
| 8.2.1  | Generic sensor cluster feature header .....              | 102 |
| 8.2.2  | Generic sensor cluster feature entity .....              | 103 |
| 8.3    | Camera feature interface.....                            | 103 |
| 8.3.1  | Camera feature header .....                              | 107 |
| 8.3.2  | Camera feature entity.....                               | 108 |
| 8.3.3  | Profile: Colour model for CFI .....                      | 109 |
| 8.4    | Ultrasonic feature interface .....                       | 110 |
| 8.4.1  | Ultrasonic feature header .....                          | 113 |
| 8.4.2  | Ultrasonic feature entity .....                          | 114 |
| 9      | Detection level.....                                     | 114 |
| 9.1    | General.....   | 114 |
| 9.2    | Generic sensor detection interface.....                  | 115 |
| 9.2.1  | Generic sensor detections header.....                    | 115 |
| 9.2.2  | Generic sensor detections entity .....                   | 116 |
| 9.3    | Radar detection interface .....                          | 117 |
| 9.3.1  | Radar detections header .....                            | 119 |
| 9.3.2  | Radar detections entity .....                            | 121 |
| 9.3.3  | Profile: Radar ambiguity .....                           | 121 |
| 9.4    | Lidar detection interface .....                          | 121 |
| 9.4.1  | Lidar detection header.....                              | 124 |
| 9.4.2  | Lidar detection entity .....                             | 125 |
| 9.5    | Camera detection interface .....                         | 125 |
| 9.5.1  | Camera detection header .....                            | 130 |
| 9.5.2  | Camera detection entity .....                            | 131 |
| 9.5.3  | Profile: Colour model for CDI.....                       | 132 |
| 9.6    | Ultrasonic detection interface.....                      | 132 |
| 9.6.1  | Ultrasonic detection header.....                         | 135 |
| 9.6.2  | Ultrasonic detection entity .....                        | 136 |
| 9.6.3  | Profile: Ultrasonic sensor cluster.....                  | 137 |
| 10     | Supportive sensor interfaces.....                        | 139 |
| 10.1   | General.....   | 139 |
| 10.2   | Generic supportive sensor interface.....                 | 140 |
| 10.2.1 | Generic supportive sensor header.....                    | 140 |
| 10.2.2 | Generic supportive sensor entity .....                   | 141 |
| 10.3   | Sensor performance interface.....                        | 141 |
| 10.3.1 | Sensor performance header.....                           | 145 |
| 10.3.2 | Sensor performance entity .....                          | 147 |
| 10.3.3 | Profile: Uniqueness of interface versioning of SPIs..... | 147 |
| 10.4   | Sensor health information interface.....                 | 147 |
| 10.4.1 | Sensor health information header.....                    | 150 |
| 10.4.2 | Sensor health information entity .....                   | 151 |
| 11     | Sensor input interface .....                             | 151 |
| 11.1   | General.....   | 151 |
| 11.2   | Generic sensor input interface.....                      | 151 |
| 11.2.1 | Generic sensor inputs header.....                        | 151 |
| 11.2.2 | Generic sensor inputs entity .....                       | 152 |
| 11.2.3 | Profile: Uniqueness of interface versioning of SII ..... | 152 |
| 11.3   | Common sensor input interface .....                      | 152 |
| 11.3.1 | Common sensor input header .....                         | 156 |
| 11.3.2 | Common sensor input entity.....                          | 156 |



|  |            |
|--|------------|
| <b>Annex A (normative) Interface signals .....</b>       | <b>158</b> |
| <b>Annex B (normative) Options and constraints .....</b> | <b>328</b> |
| <b>Bibliography .....</b>                                | <b>344</b> |

STANDARDSISO.COM : Click to view the full PDF of ISO 23150:2023

## Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see [www.iso.org/directives](http://www.iso.org/directives)).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see [www.iso.org/patents](http://www.iso.org/patents)).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see [www.iso.org/iso/foreword.html](http://www.iso.org/iso/foreword.html).

This document was prepared by Technical Committee ISO/TC 22, *Road vehicles*, Subcommittee SC 31, *Data communication*.

This second edition cancels and replaces the first edition (ISO 23150:2021), which has been technically revised.

The main changes are as follows:

- extension of the potentially moving object interface at object level (extension of the logical signal group person);
- update of the road object interface at object level (extension of the road marking sign);
- extension of the static object interface at object level (extension of the entity type traffic sign; addition of new entity type traffic sign board);
- addition of new free space area interface at object level;
- extension of the camera detection interface at detection level (addition of new entity type point; update of existing entity type shape);
- addition of new interfaces group/layer – sensor input interface (addition of new generic sensor input interface; addition of new common sensor input interface);

- extension of the error model (addition of covariances, cross-covariances and time series as error model implementation);
- refinement of the terms, for example, value as measured-, tracked- and predicted quantity value;
- new measures to link signals with their origin, that means linking signals at object level with, for example, detection entities;
- harmonisation of the document, for example, to achieve a better readability;
- update and add figures for clarification.

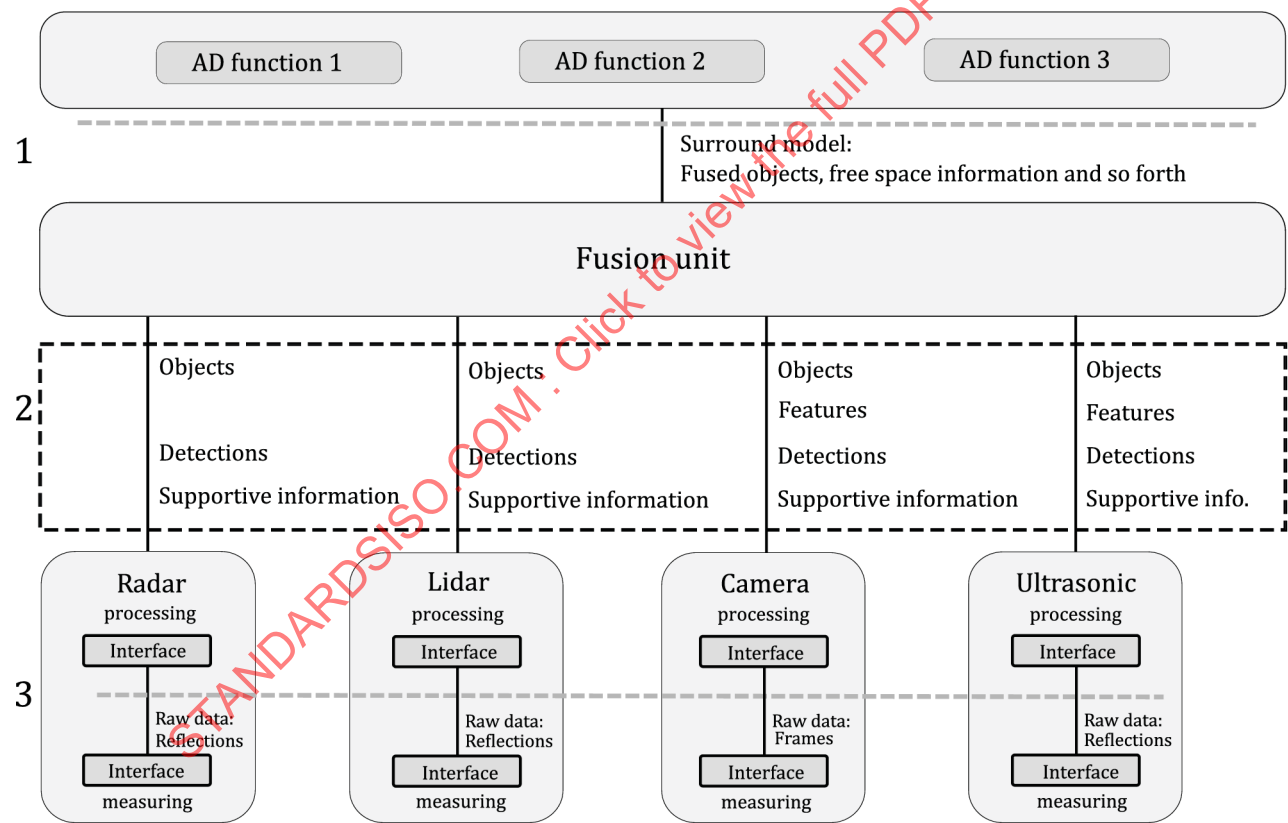
Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at [www.iso.org/members.html](http://www.iso.org/members.html).

STANDARDSISO.COM : Click to view the full PDF of ISO 23150:2023

Introduction

(Highly-)automated driving (AD) functions for road vehicles require a situation awareness of the surroundings of the vehicle and, preferably, a comprehensive scene understanding. For the fast and reliable recognition of real-world objects, a sensor suite is necessary to provide information for the fusion unit. Utilisation of different sensor technologies like radar, lidar, camera and ultrasonic with different detection capabilities is indispensable to ensure both complementary and redundant information. The fusion unit analyses and evaluates the different sensor signals and finally generates a dynamic surround model with sufficient scene understanding.

While current partly-automated functions utilise only particular objects (for example, vehicles, pedestrians, road markings) to generate a simple surround model, it is necessary for future highly-automated driving functions to merge not only the recognised objects but also to include other sensor-specific properties and characteristics of these objects for the generation of a coherent surround model of the surroundings. To minimise the development efforts for the sensors and the fusion unit and to maximise the reusability of development and validation efforts for the different functions on the sensor and fusion unit side, a standardised logical interface layer between the sensor suite and the fusion unit and a standardised logical interface layer to the sensor suite are worthwhile and beneficial for both the sensor supplier and the system supplier.

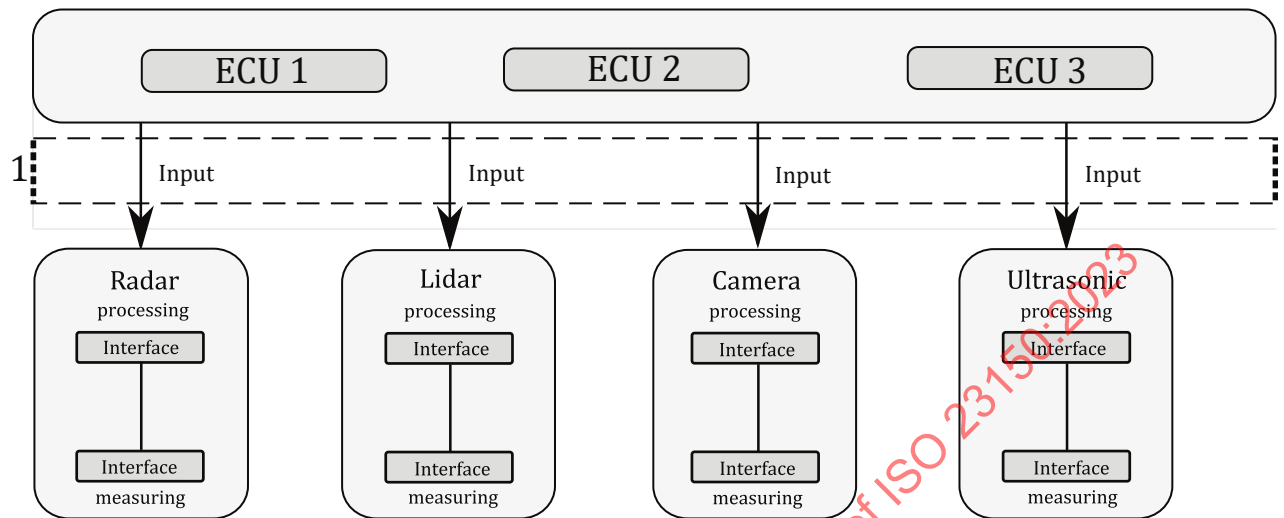


- Key**
- 1 logical interface layer between the fusion unit and AD functions
  - 2 logical interface layer between a single sensor as well as a single sensor cluster and the fusion unit
  - 3 interface layer on raw data level of a sensor's sensing element(s) and its processing

Figure 1 — Architecture: sensors/sensor clusters – fusion unit – AD functions

The logical interface layer between a single sensor as well as a single sensor cluster and the fusion unit (see Figure 1, key 2) addresses the encapsulation of technical complexity as well as objects including

free space areas, features and detections to enable object-level, feature-level and detection-level fusion. Additional supportive information of the sensor as well as the sensor cluster will supplement the data for the fusion unit.



**Key**

- 1 logical interface layer between other in-vehicle electronic control units (ECUs) (for example, odometry) and a single sensor or a single sensor cluster

**Figure 2 — Architecture: ECUs' sensor input - sensors/sensor clusters**

The logical interface layer between an electronic control unit and a single sensor as well as a single sensor cluster (see Figure 2, key 1) addresses the input of a single sensor as well as a single sensor cluster.

[STANDARDSISO.COM](https://standardsiso.com) : Click to view the full PDF of ISO 23150:2023

# Road vehicles — Data communication between sensors and data fusion unit for automated driving functions — Logical interface

## 1 Scope

This document is applicable to road vehicles with automated driving functions. The document specifies the logical interface between in-vehicle environmental perception sensors (for example, radar, lidar, camera, ultrasonic) and the fusion unit which generates a surround model and interprets the scene around the vehicle based on the sensor data. The interface is described in a modular and semantic representation and provides information on object level (for example, potentially moving objects, road objects, static objects) as well as information on feature and detection levels based on sensor technology specific information. Further supportive information is available.

This document does not provide electrical and mechanical interface specifications. Raw data interfaces are also excluded.

## 2 Normative references

There are no normative references in this document.

## 3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <http://www.electropedia.org/>

### 3.1 Architectural components

#### 3.1.1 fusion

act of uniting *signals* (3.3.1) from two or more *sensors* (3.1.5) as well as *sensor clusters* (3.1.6) to create a *surround model* (3.1.7)

#### 3.1.2 fusion unit

computing unit where the *fusion* (3.1.1) of *sensor* (3.1.5) data as well as a *sensor cluster* (3.1.6) data is performed

#### 3.1.3 interface

shared boundary between two functional units, defined by various characteristics pertaining to the functions, physical interconnections, *signal* (3.3.1) exchanges and other characteristics of the units, as appropriate

[SOURCE: ISO/IEC 2382:2015, 2124351, modified — Notes to entry have been removed.]

### 3.1.4

#### **logical interface**

*interface* (3.1.3) between a *sensor* (3.1.5) as well as a *sensor cluster* (3.1.6) and the *fusion unit* (3.1.2), defined by logical characteristics

Note 1 to entry: Logical means a semantic description of the interface.

Note 2 to entry: Mechanical and electrical interfaces are excluded.

Note 3 to entry: This document uses the term interface as a shortcut for the term logical interfaces.

### 3.1.5

#### **sensor**

in-vehicle unit which detects entities external of the vehicle with pre-processing capabilities serving at least one *logical interface* (3.1.4)

Note 1 to entry: A sensor may use one or more sensing elements.

### 3.1.6

#### **sensor cluster**

group of *sensors* (3.1.5) of the same technology serving common *logical interfaces* (3.1.4)

Note 1 to entry: A sensor cluster can exceptionally consist of only one sensor.

EXAMPLE A stereo camera, a surround-view camera, an ultrasonic sensor array, a corner radar system.

### 3.1.7

#### **surround model**

representation of the real world adjacent to the ego-vehicle

### 3.1.8

#### **in-vehicle communication**

communication network used in vehicles to connect devices to exchange information

Note 1 to entry: A in-vehicle communication connects, for example, electric control units and *sensors* (3.1.5) with each other.

## 3.2 Terms for logical interface layers

### 3.2.1

#### **detection**

sensor technology specific entity represented in the *sensor coordinate system* (3.7.18) based on a single *measurement* (3.4.5) of a *sensor* (3.1.5)

Note 1 to entry: A small amount of history can be used for some detection *signals* (3.3.1), for example, model-free filtering may be used in track-before-detect algorithms.

### 3.2.2

#### **detection level**

set of *logical interfaces* (3.1.4) that provides *detections* (3.2.1)

### 3.2.3

#### **feature**

sensor technology specific entity represented in the *vehicle coordinate system* (3.7.16) based on multiple *measurements* (3.4.5)



Note 1 to entry: Multiple measurements can originate from a *sensor cluster* (3.1.6).

Note 2 to entry: Multiple measurements can originate from multiple *measurement cycles* (3.4.1).

Note 3 to entry: The term *feature* is used in this document not as function or group of functions as specified in ISO/SAE PAS 22736:2021.

### 3.2.4

#### **feature level**

set of *logical interfaces* (3.1.4) that provides *features* (3.2.3)

### 3.2.5

#### **object**

representation of a real-world entity with defined boundaries and characteristics in the *vehicle coordinate system* (3.7.16)

Note 1 to entry: The geometric description of the object is in the vehicle coordinate system.

Note 2 to entry: Object *signals* (3.3.1) are basically sensor technology independent. Sensor technology specific signals may extend the object signals.

EXAMPLE A *potentially moving object* (3.6.1), a *road object* (3.6.2), a *static object* (3.6.3), a *free space area object* (3.6.4).

### 3.2.6

#### **object level**

set of *logical interfaces* (3.1.4) that provides *objects* (3.2.5)

### 3.2.7

#### **sensor input**

data received by a *sensor* (3.1.5) or a *sensor cluster* (3.1.6) via the *in-vehicle communication* (3.1.8)

## 3.3 Structure terms

### 3.3.1

#### **signal**

entity consisting of one or more values and which is part of a *logical interface* (3.1.4)

### 3.3.2

#### **logical signal group**

grouping of *signals* (3.3.1) that has a logical relationship and a name for the grouping

### 3.3.3

#### **classification**

attribute-based differentiation

Note 1 to entry: An attribute is defined by a list of enumerators.

## 3.4 Measurement terms

### 3.4.1

#### **measurement cycle**

time period from the start of a data acquisition event to the start of the next data acquisition event

Note 1 to entry: A measurement cycle of one *sensor* (3.1.5) is a consistent view of an observed scene and not overlapping in time.

### 3.4.2

#### **measured quantity value**

value of a quantity resulting from a *measurement* (3.4.5)

### 3.4.3

#### **tracked quantity value**

value of a quantity determined from observed sequential changes, using information related to the same characteristic

### 3.4.4

#### **predicted quantity value**

value of a quantity assessed before it is actually observable, using information related to the same characteristic

EXAMPLE Related information can be recent and previous *measured quantity values* (3.4.2), *tracked quantity values* (3.4.3) and state variables.

[SOURCE: IEV 192-13-02, modified — EXAMPLE has been added and the word "quantity" has been added to the term.]

### 3.4.5

#### **measurement**

processing result of a *measurement cycle* (3.4.1)

### 3.4.6

#### **tracking**

computation process used to calculate the *tracked quantity value* (3.4.3) of a quantity

### 3.4.7

#### **prediction**

computation process used to obtain the *predicted quantity value* (3.4.4) of a quantity

[SOURCE: IEV 192-11-01]

### 3.4.8

#### **error**

discrepancy between a *measured quantity value* (3.4.2), *tracked quantity value* (3.4.3) or *predicted quantity value* (3.4.4) or condition, and the true, specified or theoretically correct reference quantity value or condition

Note 1 to entry: An error within a system can be caused by failure of one or more of its components, or by the activation of a systematic fault.

[SOURCE: IEV 192-03-02, modified — "computed, observed or measured value" was modified to "measured quantity value, tracked quantity value or predicted quantity value", "value" was modified to "reference quantity value", Note 1 to entry has been adapted and Note 2 to entry was deleted.]

### 3.4.9

#### **accuracy**

closeness of agreement between a *measured quantity value* (3.4.2), *tracked quantity value* (3.4.3) or *predicted quantity value* (3.4.4) and a true quantity value

Note 1 to entry: The concept accuracy is not a quantity and is not given a numerical quantity value. A *measurement* (3.4.5), *tracking* (3.4.6) or *prediction* (3.4.7) is said to be more accurate when it offers a smaller *error* (3.4.8).

Note 2 to entry: The term accuracy should not be used for *trueness* (3.4.10) and the term *precision* (3.4.11) should not be used for accuracy, which, however, is related to both these concepts.

Note 3 to entry: Accuracy is sometimes understood as closeness of agreement between measured, tracked or predicted quantity values that are being attributed to the measurand.

[SOURCE: ISO/IEC Guide 99:2007, 2.13, modified — The terms "measurement accuracy" and "accuracy of measurement" were deleted, definition was extended for tracked or predicted quantity values and the Notes to entry have been adapted.]

### 3.4.10

#### **trueness**

closeness of agreement between the average of an infinite number of replicated *measured quantity values* (3.4.2), *tracked quantity values* (3.4.3) or *predicted quantity values* (3.4.4) and a reference quantity value

Note 1 to entry: Trueness is not a quantity and thus cannot be expressed numerically, but measures for closeness of agreement are given in the ISO 5725 series.

Note 2 to entry: Trueness is inversely related to systematic error but is not related to random error.

Note 3 to entry: The term *accuracy* (3.4.9) should not be used for trueness.

[SOURCE: ISO/IEC Guide 99:2007, 2.14, modified — The terms "measurement trueness" and "trueness of measurement" were deleted, definition was extended for tracked or predicted quantity values and the Notes to entry have been adapted.]

### 3.4.11

#### **precision**

closeness of agreement between indications or *measured quantity values* (3.4.2), *tracked quantity value* (3.4.3) or *predicted quantity values* (3.4.4) obtained by replicate *measurements* (3.4.5), *tracking* (3.4.6) or *prediction* (3.4.7) on the same or similar measurands under specified conditions

Note 1 to entry: Precision is usually expressed numerically by measures, trackings or predictions of imprecision, such as standard deviation, variance or coefficient of variation under the specified conditions of measurement, tracking or prediction.

Note 2 to entry: The specified conditions can be, for example, repeatability conditions of measurement, intermediate precision conditions of measurement or reproducibility conditions of measurement (see ISO 5725-1:1994).

Note 3 to entry: Precision is used to define measurement, tracking or prediction repeatability, intermediate measurement or prediction precision and measurement, tracking or prediction reproducibility.

Note 4 to entry: Sometimes precision is erroneously used to mean *accuracy* (3.4.9).

Note 5 to entry: Precision is inversely related to random error but is not related to systematic error.

[SOURCE: ISO/IEC Guide 99:2007, 2.15, modified — The term "measurement precision" was deleted, the word "objects" was replaced by "measurands", definition was extended for tracked or predicted quantity values, the Notes to entry have been adapted and Note 5 to entry has been added.]

### 3.4.12

#### **measurement error**

*measured quantity value* (3.4.2) minus a reference quantity value

Note 1 to entry: The concept of measurement error can be used both:

- a) when there is a single reference quantity value to refer to, which occurs if a calibration is made by means of a measurement standard with a measured quantity value having a negligible measurement uncertainty or if a conventional quantity value is given, in which case the *error* (3.4.8) is known, and
- b) if a measurand is supposed to be represented by a unique true quantity value or a set of true quantity values of negligible range, in which case the error is not known.

Note 2 to entry: Measurement error should not be confused with production error or mistake.

[SOURCE: ISO/IEC Guide 99:2007, 2.16, modified — The terms "error" and "error of measurement" were deleted and the Notes to entry have been adapted.]

### 3.4.13

#### **tracking error**

quantitative statement about the *tracked quantity value* (3.4.3) and the reference quantity value

### 3.4.14

#### **prediction error**

quantitative statement about the *predicted quantity value* (3.4.4) and the reference quantity value

### 3.4.15

#### **error model**

model used to estimate the *error* (3.4.8)

### 3.4.16

#### **fixation**

short temporal holds of movements that keep alignment of the eyes to a particular point within an area of interest which falls on the fovea (the middle of the retina responsible for our central, sharpest vision) for a given time period

[SOURCE: ISO 15007:2020, 3.1.4, modified — Notes to entry have been deleted.]

## 3.5 Requirement level terms

### 3.5.1

#### **conditional**

required under certain specified conditions

Note 1 to entry: One of three obligation statuses applied to a *requirement level* (3.5.4) of a *logical interface* (3.1.4) specification, indicating the conditions under which the *signal* (3.3.1) or *logical signal group* (3.3.2) is required. In other cases, the signal or logical signal group is optional. See also *mandatory* (3.5.2) and *optional* (3.5.3).

[SOURCE: ISO/IEC 11179-3:2023, 3.2.77, modified — Note 1 to entry has been adapted and Note 2 to entry has been removed.]

### 3.5.2

#### **mandatory**

always required

Note 1 to entry: One of three obligation statuses applied to a *requirement level* (3.5.4) of a *logical interface* (3.1.4) specification, indicating the conditions under which the *signal* (3.3.1) or *logical signal group* (3.3.2) is required. See also *conditional* (3.5.1) and *optional* (3.5.3).

[SOURCE: ISO/IEC 11179-3:2023, 3.2.75, modified — Note 1 to entry has been adapted and Note 2 to entry has been removed.]

### 3.5.3

#### **optional**

permitted but not required

Note 1 to entry: One of three obligation statuses applied to a *requirement level* (3.5.4) of a *logical interface* (3.1.4) specification, indicating the conditions under which the *signal* (3.3.1) or *logical signal group* (3.3.2) is required. See also *conditional* (3.5.1) and *mandatory* (3.5.2).

[SOURCE: ISO/IEC 11179-3:2023, 3.2.76, modified — Note 1 to entry has been adapted and Note 2 to entry has been removed.]

### 3.5.4

#### **requirement level**

definition of the obligation status of a *logical interface's* (3.1.4) *logical signal group* (3.3.2), *signal* (3.3.1) as well as a signal's identifier or signal's enumerator

Note 1 to entry: Each requirement level entry has one of three possible obligation statuses applied: *conditional* (3.5.1), *mandatory* (3.5.2) or *optional* (3.5.3).

## 3.6 Road user relevant entity types

### 3.6.1

#### **potentially moving object**

real-world entity which can potentially move and is relevant for driving situations

Note 1 to entry: A representation of a potentially moving object is part of *logical interfaces* (3.1.4) on *object level* (3.2.6).

EXAMPLE A vehicle, a bicycle, a pedestrian, an obstacle.

### 3.6.2

#### **road object**

marking or structure of a road which is relevant for driving situations

Note 1 to entry: A representation of a road object is part of *logical interfaces* (3.1.4) on *object level* (3.2.6).

EXAMPLE A *road marking* (3.6.2.1), a *road boundary* (3.6.2.2), the *road surface* (3.6.2.3).

#### 3.6.2.1

##### **road marking**

line, symbol or other mark on the surface of a road or a structure intended to limit, regulate, warn, guide or inform road users

Note 1 to entry: Other marks could be text, numbers, arrows or combinations.

EXAMPLE A lane marking, Botts' dots.

[SOURCE: ISO 6707-1:2020, 3.3.5.80, modified — "user" was modified to "road users", "limit" was added and the Note 1 to entry and EXAMPLE have been added.]

#### 3.6.2.2

##### **road boundary**

structure that limits the road

EXAMPLE A curb stone, a guard rail, the end of the surface of the road.

### 3.6.2.3

#### **road surface**

surface supporting the tyre and providing friction necessary to generate shear forces in the *road plane* (3.7.6)

Note 1 to entry: The surface may be flat, curved, undulated or of other shape.

[SOURCE: ISO 8855:2011, 2.6]

### 3.6.3

#### **static object**

real-world stationary entity which can be used for information and/or localisation

Note 1 to entry: A representation of a static object is part of *logical interfaces* (3.1.4) on *object level* (3.2.6).

EXAMPLE A *general landmark* (3.6.3.1), a *traffic sign* (3.6.3.2), a *traffic sign board* (3.6.3.3), a *traffic light* (3.6.3.4).

#### 3.6.3.1

##### **general landmark**

real-world stationary entity which can be used for localisation

Note 1 to entry: A stationary *traffic sign* (3.6.3.2) or *traffic light* (3.6.3.4) is also regarded as a general landmark.

EXAMPLE A building, a tunnel, a bridge, a sign gantry structure, a tree.

#### 3.6.3.2

##### **traffic sign**

traffic relevant, authorised sign that limits, regulates, warns, guides or informs road users

Note 1 to entry: One traffic sign usually consists of one *main sign* (3.6.3.2.1) and none, one or several *supplementary signs* (3.6.3.2.2).

EXAMPLE A speed limit which is restricted for trucks.

##### 3.6.3.2.1

###### **main sign**

*traffic sign* (3.6.3.2) which gives a general message, obtained by a combination of colour and geometric shape and which, by the addition of a graphical symbol or text, gives a particular message for road users

[SOURCE: ISO 3864-1:2011, 3.12, modified — The original term was "safety sign", "sign" has been replaced by "traffic sign" and the phrases "or text" and "for road users" have been added to the definition.]

##### 3.6.3.2.2

###### **supplementary sign**

*traffic sign* (3.6.3.2) that is supportive of a *main sign* (3.6.3.2.1) and the main purpose of which is to provide additional clarification

[SOURCE: ISO 3864-1:2011, 3.14, modified — "traffic sign" now replaces "sign" and "main sign" replaces "traffic sign".]

#### 3.6.3.3

##### **traffic sign board**

traffic relevant, authorised sign board that limits, regulates, warns, guides or informs road users by abstract representations of lanes and streets ahead to control traffic on the road

#### 3.6.3.4

##### **traffic light**

traffic relevant, official lights

Note 1 to entry: One traffic light consists of one or several light spots with different light colours and/or shapes.

EXAMPLE A pedestrian traffic light.

#### 3.6.4

##### **free space area object**

area entity of the *road surface* (3.6.2.3) that can be driven on and is free of obstacles that can limit driving

Note 1 to entry: A representation of a free space area object is part of *logical interfaces* (3.1.4) on *object level* (3.2.6).

### 3.7 Axis and coordinate system terms

#### 3.7.1

##### **reference frame**

geometric environment in which all points remain fixed with respect to each other at all times

[SOURCE: ISO 8855:2011, 2.1]

#### 3.7.2

##### **axis system**

set of direction vectors

#### 3.7.3

##### **coordinate system**

numbering convention used to assign a unique ordered trio of values to each point in a *reference frame* (3.7.1) and which consists of an *axis system* (3.7.2) plus an origin point

[SOURCE: ISO 8855:2011, 2.4, modified — "(x, y, z)" has been removed from the definition.]

#### 3.7.4

##### **cartesian coordinate system**

set of numerical coordinates (x, y, z), which are the signed distances to the YZ-, ZX- and XY-planes using a set of three orthogonal directions associated with X, Y and Z axes

Note 1 to entry: A right-handed *axis system* (3.7.2) is assumed throughout this document, where:  $\vec{Z} = \vec{X} \times \vec{Y}$ .

#### 3.7.5

##### **spherical coordinate system**

set of one distance vector and two angles associated with radial distance, azimuth and elevation

Note 1 to entry: The azimuth angle is the angle in XY-plane [of a *cartesian coordinate system* (3.7.4)] of the *axis system* (3.7.2) counted from the X-axis. The elevation angle is the angle from the azimuth direction in the XY-plane of the axis system towards the direction of the distance vector, that is XY-plane has an elevation angle = 0 rad.

Note 2 to entry: The angles of the spherical coordinate system have increasing values in counterclockwise direction.

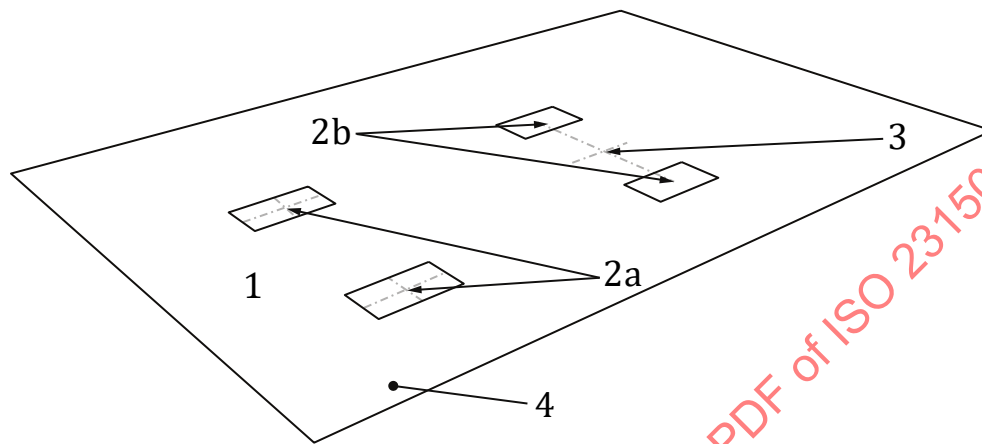
### 3.7.6

#### road plane

plane representing the *road surface* (3.6.2.3) within the front tyre contact patches and the *vehicle road-level reference point* (3.7.13)

Note 1 to entry: See Figure 3.

Note 2 to entry: For tyre contact patches, see ISO 8855:2011, 4.1.5.



#### Key

- 1 vehicle front
- 2a vehicle's front tyre contact patches
- 2b vehicle's rear tyre contact patches
- 3 vehicle road-level reference point (3.7.13)
- 4 vehicle road plane

**Figure 3 — Road plane**

[SOURCE: ISO 8855:2011, 2.7, modified — "tyre contact patch" was modified to "front tyre contact patches and the vehicle road-level reference point", the figure has been added and the Notes to entry have been modified.]

### 3.7.7

#### road level

point related to a *road plane* (3.7.6)

### 3.7.8

#### vehicle unsprung mass

unsprung mass

mass that is not carried by the suspension, but is supported directly by the tyres

[SOURCE: ISO 8855:2011, 4.11, modified — The term "vehicle unsprung mass" has been added.]

### 3.7.9

#### vehicle sprung mass

sprung mass

mass that is supported by the suspension, that is the total vehicle mass less the *vehicle unsprung mass* (3.7.8)

[SOURCE: ISO 8855:2011, 4.12, modified — The term vehicle sprung mass has been added and Note 1 to entry has been removed.]



**3.7.10****vehicle rear-axle reference point**

point fixed in the *vehicle sprung mass* (3.7.9) and located at the centre of the rear-axle

**3.7.11****vehicle sprung mass axis system**

*axis system* (3.7.2) fixed in the *reference frame* (3.7.1) of the *vehicle sprung mass* (3.7.9), so that the *X*-axis is substantially horizontal and forwards (with the vehicle at rest) and is parallel to the vehicle's longitudinal plane of symmetry, additionally the *Y*-axis is perpendicular to the vehicle's longitudinal plane of symmetry and points to the left with the *Z*-axis pointing upward

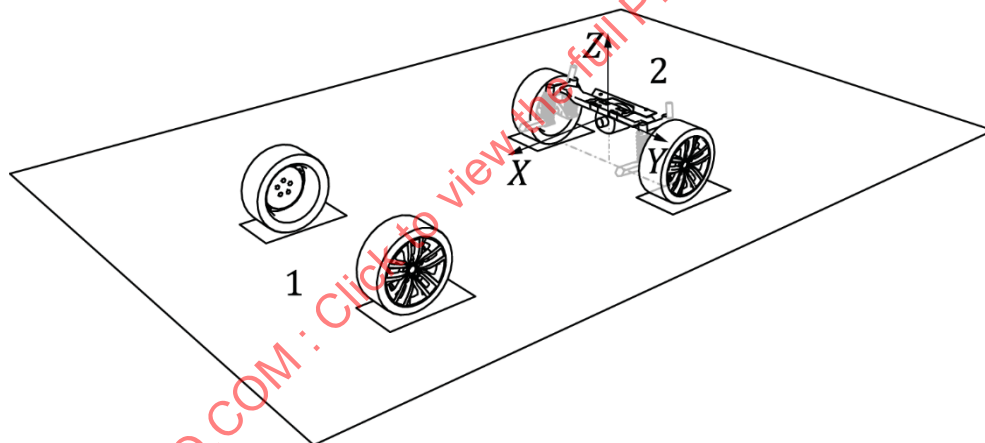
Note 1 to entry: The axis system is a set of three orthogonal directions associated with *X*, *Y* and *Z* axes.

**3.7.12****vehicle rear-axle coordinate system**

*cartesian coordinate system* (3.7.4) based on the *vehicle sprung mass axis system* (3.7.11) with the origin located at the *vehicle rear-axle reference point* (3.7.10)

Note 1 to entry: The vehicle rear-axle coordinate system is a *vehicle coordinate system* (3.7.16).

Note 2 to entry: See Figure 4.

**Key**

- 1 vehicle front
- 2 vehicle rear-axle reference point (3.7.10)

**Figure 4 — Vehicle rear-axle coordinate system**

**3.7.13****vehicle road-level reference point**

point at *road level* (3.7.7) located in the middle of the rear tyre contact patches

Note 1 to entry: For tyre contact patches, see ISO 8855:2011, 4.1.5.

**3.7.14****vehicle road-level axis system**

*axis system* (3.7.2) fixed in the *reference frame* (3.7.1) of the *vehicle unsprung mass* (3.7.8), so that the *X*-axis is parallel to the vehicle's longitudinal plane of symmetry and points into forward moving direction and the *Y*-axis is perpendicular to the vehicle's longitudinal plane of symmetry and points to the left with the *Z*-axis pointing upward

Note 1 to entry: Vehicle road-level axis system's  $XY$ -plane is parallel to the ego-vehicle's *road plane* (3.7.6).

Note 2 to entry: The axis system is a set of three orthogonal directions associated with  $X$ ,  $Y$  and  $Z$  axes.

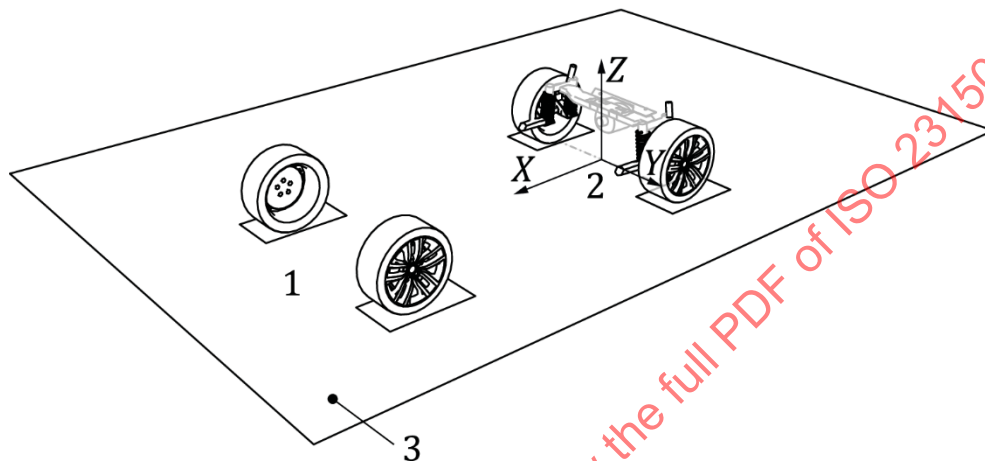
### 3.7.15

#### vehicle road-level coordinate system

*cartesian coordinate system* (3.7.4) based on the *vehicle road-level axis system* (3.7.14) with the origin located at the *vehicle road-level reference point* (3.7.13) at the *vehicle road level* (3.7.7)

Note 1 to entry: The vehicle road-level coordinate system is a *vehicle coordinate system* (3.7.16).

Note 2 to entry: See Figure 5.



#### Key

- 1 vehicle front
- 2 vehicle road-level reference point (3.7.13)
- 3 vehicle road plane (3.7.6)

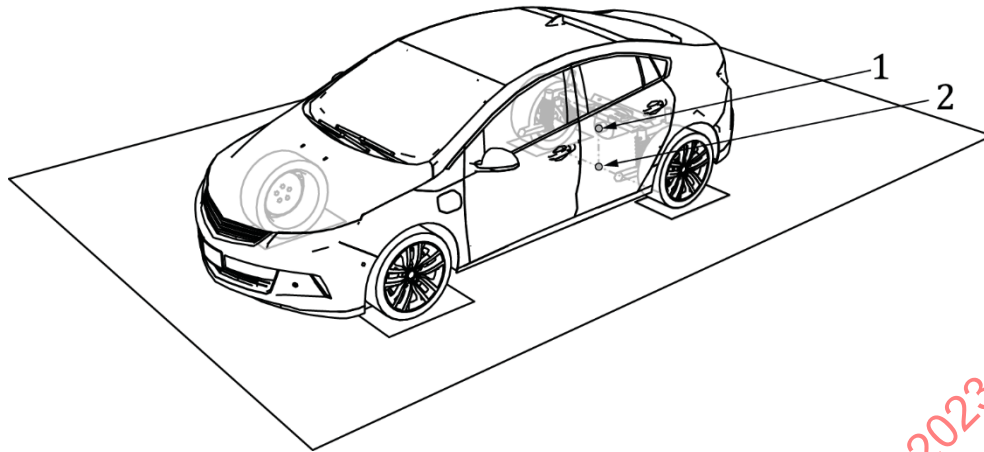
**Figure 5 — Vehicle road-level coordinate system**

### 3.7.16

#### vehicle coordinate system

*cartesian coordinate system* (3.7.4) which is either the *vehicle rear-axle coordinate system* (3.7.12) or the *vehicle road-level coordinate system* (3.7.15)

Note 1 to entry: See Figure 6.

**Key**

- 1 vehicle rear-axle reference point (3.7.10) of the vehicle rear-axle coordinate system (3.7.12)
- 2 vehicle road-level reference point (3.7.13) of the vehicle road-level coordinate system (3.7.15)

**Figure 6 — Vehicle coordinate systems****3.7.17****sensor axis system**

*axis system* (3.7.2) fixed in the *reference frame* (3.7.1) of the *sensor* (3.1.5)

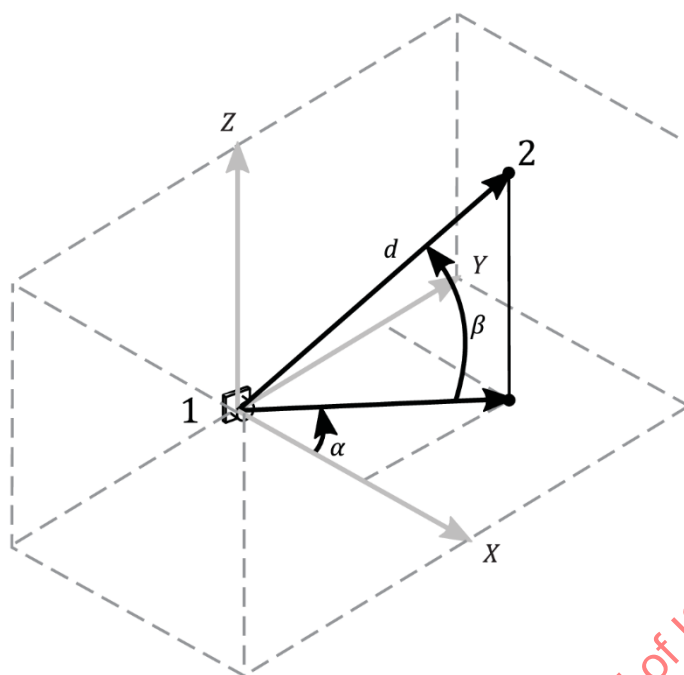
Note 1 to entry: The *X*-axis [of a *cartesian coordinate system* (3.7.4)] is in viewing direction of the sensor and the *Z*-axis pointing upward.

**3.7.18****sensor coordinate system**

*spherical coordinate system* (3.7.5) based on the *sensor axis system* (3.7.17) at a defined origin point of the *sensor* (3.1.5)

Note 1 to entry: The origin point of the sensor coordinate system has to be selected in a way that *detections* (3.2.1) could easily be specified in a spherical coordinate system. For example, the origin point of a camera sensor is the virtual projection centre of the camera's optics.

Note 2 to entry: See Figure 7.



**Key**

- 1 *sensor* (3.1.5) with origin point of the sensor coordinate system
- 2 *detection* (3.2.1)
- $d$  radial distance
- $\alpha$  azimuth angle
- $\beta$  elevation angle

**Figure 7 — Sensor coordinate system**

**3.7.19**

**affine coordinate system**

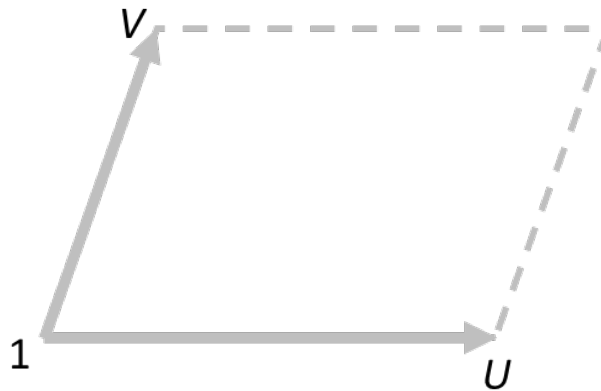
*coordinate system* (3.7.3) of an *object* (3.2.5) with an ordered set of non-collinear direction vectors (an affine basis) and a defined origin point

Note 1 to entry: The length of the direction vectors may not be normalised.

Note 2 to entry: For a planar affine coordinate system of an object, a pair of direction vectors (abscissa and ordinate) and an origin point are necessary to define the affine coordinate system.

Note 3 to entry: A right-handed *axis system* (3.7.2) is assumed for a three-dimensional affine coordinate system (with the direction vectors abscissa, ordinate and applicate).

Note 4 to entry: See Figure 8.

**Key**

- 1 origin point of the affine coordinate system  
 U U-axis – abscissa of the affine coordinate system  
 V V-axis – ordinate of the affine coordinate system

**Figure 8 — Affine coordinate system****4 Abbreviated terms**

|       |   |
|-------|---|
| A2I   | assign to interface                     |
| AD    | automated driving                       |
| AER   | alternative entity representation       |
| C     | conditional                             |
| CDI   | camera detection interface              |
| CFI   | camera feature interface                |
| CSII  | common sensor input interface           |
| D     | dimensional                             |
| DLI   | detection level interface               |
| ECU   | electronic control unit                 |
| EM    | error model                             |
| FFT   | fast Fourier transform                  |
| FLI   | feature level interface                 |
| FLP   | front-left position of the ego-vehicle  |
| FOV   | field-of-view                           |
| FRP   | front-right position of the ego-vehicle |
| FSAOI | free space area object interface        |
| FWHM  | full width at half maximum              |
| GPS   | global positioning system               |
| HW    | hardware                                |
| ID    | identifier                              |
| IQR   | interquartile range                     |

|      |  |
|------|--|
| IR   | infrared                               |
| IRI  | international roughness index          |
| IV   | implicit value                         |
| LDI  | lidar detection interface              |
| LL   | list length                            |
| LSG  | logical signal group                   |
| M    | mandatory                              |
| NIR  | near infrared                          |
| O    | optional                               |
| OLI  | object level interface                 |
| PMOI | potentially moving object interface    |
| RCS  | radar cross section                    |
| RDI  | radar detection interface              |
| RDOI | road object interface                  |
| RL   | requirement level                      |
| RLP  | rear-left position of the ego-vehicle  |
| RRP  | rear-right position of the ego-vehicle |
| SHII | sensor health information interface    |
| SII  | sensor input interface                 |
| SNR  | signal to noise ratio                  |
| SOI  | static object interface                |
| SPI  | sensor performance interface           |
| SSI  | supportive sensor interface            |
| UDI  | ultrasonic detection interface         |
| UFI  | ultrasonic feature interface           |
| UTL  | unrolling tuple list                   |
| VRO  | value representation optimisation      |

## **5 Structure of the interface description**

### **5.1 General**

The scope of this document are logical interfaces of sensors and sensor clusters. Therefore, the following parts of this document use the term interface as a shortcut for the defined term “logical interface” (3.1.4). The interface descriptions have the following structure:

- a description of each interface by logical signal group (LSGs), signal names and requirement levels (RLs) (including conditional requirements) and additional options;
- a detailed description of each signal in Annex A;
- generic options and generic conditional requirements in Annex B, which shall be used.

The requirement level attribute “conditional” shall be an element used in this document. The logical interlink as well as complex dependencies between signals of an interface shall be defined in an interface’s subclause as a profile subclause of the interface.

## 5.2 Signal

Table 1 provides an overview of a signal used by an interface which is defined in subclause “Interface” (5.3). All signals are specified in Annex A. Signal names shall be unique. Similar signals on different interface levels have the interface level added to the signal name to guarantee a unique signal naming.

**Table 1 — Signal: <Signal name>**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Name of the signal and optional list of signal identifiers<br>Each signal respectively signal identifier is implemented by exactly one value.   |             |   |
| <b>Description</b> | Definition of the signal  |             |   |
| <b>Value type</b>  | Definition of the signal’s value type and optionally the number of values, specifically the number of dimensions defined by the number of identifiers<br>(default dimension: 1D vector) | <b>Unit</b> | Unit definition of the signal value<br>In this document, the unit defines no application or network representation of the signal value (for example, a scaling factor, a fix point representation, linear or non-linear increasing values). |

In general, for each signal with value type “enumeration”, a list of enumerators or exemplary enumerators is defined in an additional table (see Table 2) in Annex A. Each list of exemplary enumerators is informative. During the system design phase, specific enumerators shall be defined, based upon the list of exemplary enumerators (see Annex A). This can be done by adding new, using a subset of or adapting the existing exemplary enumerators. The implementation of a list of exemplary enumerators shall contain at least one enumerator and should have default ones (B.1.7). An enumeration with exemplary enumerators may allow output of multiple enumerators belonging to different classes. Enumerators of different classes may be otherwise specified during the system design phase by separated enumeration signal identifiers.

Each enumeration with exemplary enumerators shall implement default enumerators (B.1.7). At least one enumerator shall be defined for each implemented exemplary enumeration list. An enumerator with exemplary enumerations may be clustered by multiple enumerations to deal with different classes of enumerators, rather than addressing multiple classes in the enumerators.

**Table 2 — Enumeration: <Signal name>**

|  |                              |   |
|--|------------------------------|---|
| <b>Name</b>  | <b>Description</b>           | <b>RL enumerator</b>  |
| Name of the enumerator<br>(the signal name is abbreviated in the name of the enumerator) | Definition of the enumerator | Requirement level of the enumerator<br>(mandatory, conditional or optional)<br>or “exemplary” |

## 5.3 Interface

Table 3 provides an overview of a logical interface. An interface is comprised of signals. Signals can be part of a logical signal group (LSG) for ease of logical grouping.

Table 3 — Interface: &lt;Interface name&gt;

| LSG  | RL LSG<br>M/C/O   | Signal   | RL signal<br>M/C/O   | Option   |
|--|---|--|--|--|
| Name of the logical signal group (LSG) – for a logical group of multiple signals | Requirement level for LSG: <b>mandatory</b> , <b>conditional</b> or <b>optional</b><br><br>Conditional RL with a reference to a clause in Annex B | Name of the signal, optional: {list of signal identifiers} with a reference to a clause in Annex A | Requirement level for signal: <b>mandatory</b> , <b>conditional</b> or <b>optional</b> is specified for the signal, including all signal identifiers, or for each signal identifier individually<br><br>Conditional RL with a reference to a clause in Annex B | Additional options for the signal:<br>— a reference to a clause in Annex B<br>— a reference to a profile subclause |

Areas in the interface table marked in grey indicate that the enclosed signals are part of a list of entities. The valid number of entities is defined at the beginning of the grey area (specifically, grey area header).

If an interface provides multiple interface entity types (for example, the road object interface [RDOI] provides road surface, road markings and road boundaries), the interface may require not all entity types at once (see specified in B.2.3). Different entry types of an interface are separated with a double horizontal line.

A conditional LSG requirement level shall define the conditions. Under all other conditions, the LSG is optional [see the definition conditional (3.5.1)].

An LSG shall have at least one signal that is mandatory. If a signal of an optional LSG is used, all signals of the LSG shall be considered as defined on their signal requirement levels.

In case of a list, the requirement level of the signal “Number of valid <...>” and its LSG are the base requirement level of all signals in the list.

Symbol “” indicates that LSG belongs to another LSG, that means the LSG is a nested LSG.

If only one signal requirement level is defined for a signal with signal identifiers, the requirement level of each signal identifier individually corresponds to the requirement level for the signal.

Signals with RL mandatory and static or fixed values, which are defined during the system design phase and/or are implicitly known by the sensor and the fusion unit may be optimised. In this case the applicability of optimisation methods of B.1 are allowed which essentially means that, for example, the mandatory signals will not be explicitly transferred by the interface message. During the system design phase, additional measures from Annex B are applicable and shall be defined for the sensor as well as the sensor cluster.

## 5.4 Specific signal grouping


Table 4 provides an overview of a specific signal grouping, for example, an LSG, which refers a specific signal grouping of another subclause and extends the specific signal grouping to achieve similar interfaces.

Table 4 — Specific signal grouping: &lt;Name for LSG(s)&gt;

| Signal | RL signal | Option |
|--------|-----------|--------|
|--------|-----------|--------|



|   |   |   |
|---|---|---|
| Name of the signal, optional:<br>{list of signal identifiers} with<br>a reference to a clause in<br>Annex A | Requirement level for signal: mandatory,<br>conditional or optional is specified for the<br>signal, including all signal identifiers, or for<br>each signal identifier individually<br>Conditional RL with a reference to a clause<br>in Annex B<br>Optional: information of the RL for the<br>signal's LSG, if it is not mandatory | Additional options for the<br>signal:<br>— a reference to a clause in<br>Annex B<br>— a reference to a profile<br>subclause |
|---|---|---|

Symbol “” indicates that the signal belongs to a list of signals.

Unchanged, inherited signals, signal requirement levels and signal options of the referenced subclause's specific signal grouping are marked in grey.

## 5.5 Profile

If multiple signals of an interface have an interface-specific semantic interdependency, this interdependency is described in a profile subclause and additional complex conditions are defined. Each signal which is contained in the profile has an annotation of the profile in the signal's option-column definition of the interface (see column “Option” in Table 3 and Table 4).

## 6 Logical interface from a sensor as well as a sensor cluster to a fusion unit

### 6.1 General

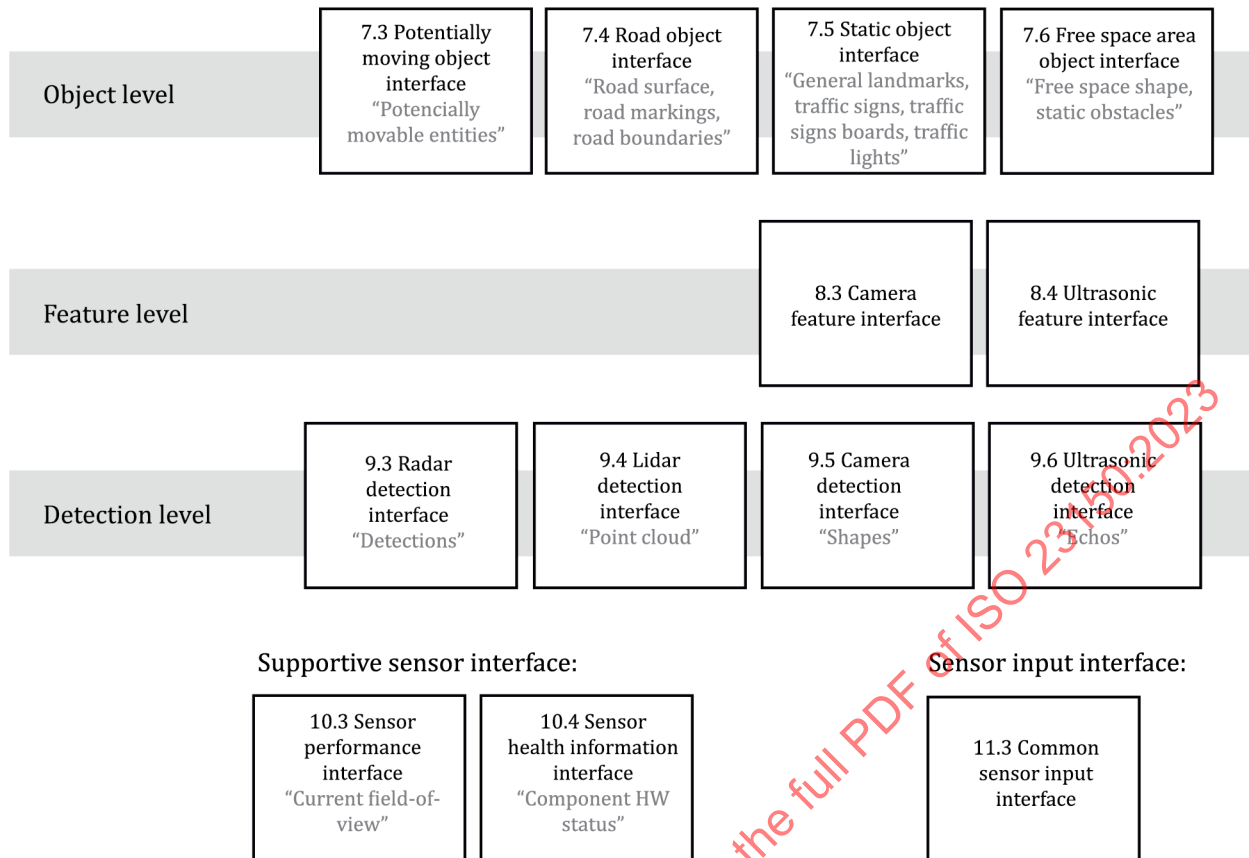
The aim of this document is to support different levels of fusion. Therefore, the document provides three logical interface levels (detection-, feature- and object level). Supportive sensor interfaces (SSIs) complement the interfaces of the three logical interface levels. Detections, features, objects and supportive sensor information are the basis for a fusion unit. A tracking algorithm within the sensor solely or the sensor cluster using their own information, results in objects with limited quality. The fusion unit can improve these and is able to merge the information on different levels, using object-level fusion as well as feature- and/or detection-level fusion. In order to perform object-level fusion from a sensor suite<sup>1</sup>, sufficient evidence in the form of correlated detections belonging to a particular real-world entity shall be provided by each sensor and sensor cluster. An advantage of feature-level fusion is the possible creation of an object in the common fusion unit out of detections, features or objects coming from a sensor suite, which would not have resulted in an object recognition within each sensor or each sensor cluster itself. Detection-level fusion may use machine learning algorithms to handle complexity. Future vehicles will integrate a sensor suite with many sensors and sensor clusters using different sensor technologies. Different technologies will complement each other to achieve a robust and reliable fusion.

Supportive sensor interfaces are provided so that the fusion is able to process the reliability of the used measurement method or data. These interfaces are sensor technology independent, specifically there are uniform interfaces for various sensor technologies. These supportive sensor interfaces are also used for the implementation of safety concepts.

Sensor input interfaces are used to provide data and parametrise the sensors or the sensor cluster from other devices via the in-vehicle communication.

Figure 9 shows the interfaces that are in the scope of this document.

<sup>1</sup> It consists of two or more different and independent sensors or sensor clusters.



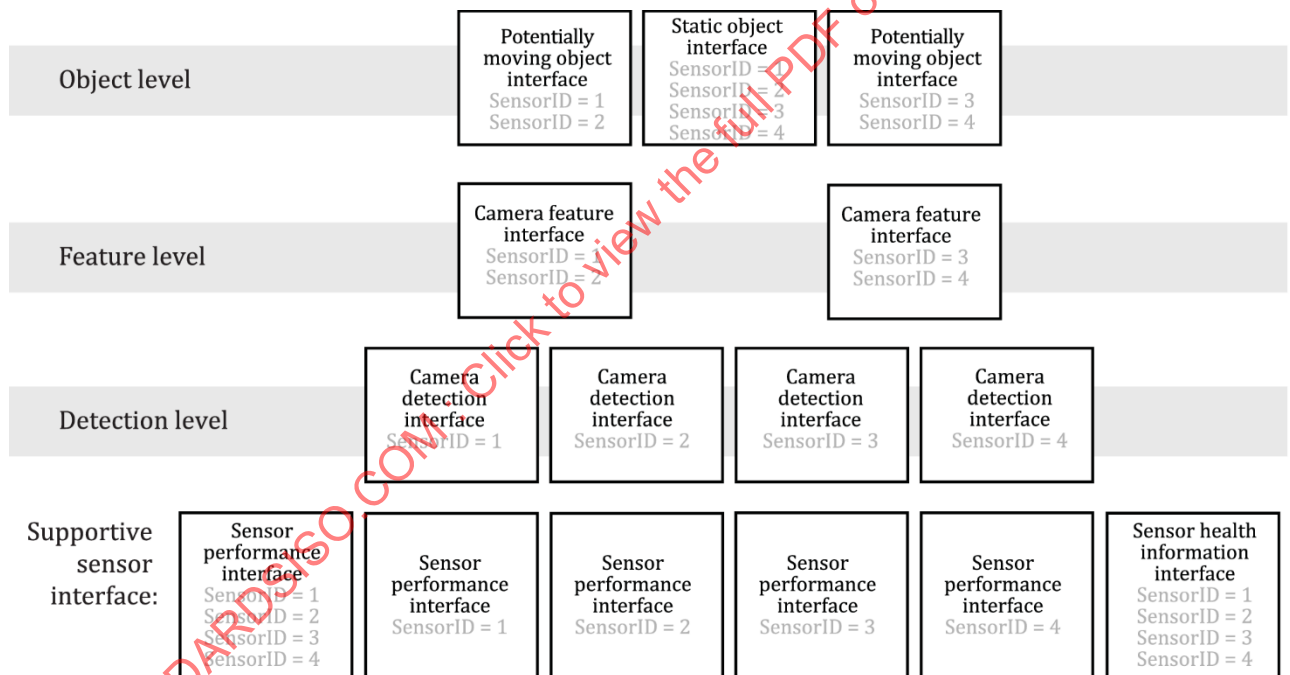
**Figure 9 — Interfaces in the scope of this document**

The interfaces covered in the scope of this document are the following:

- objects:
  - Potentially moving object interface (7.3): with specific attributes for, for example, lights, pedestrian;
  - Road object interface (7.4): road surface, road markings, road boundaries;
  - Static object interface (7.5): general landmarks, traffic signs, traffic sign boards, traffic lights;
  - Free space area object interface (7.6): free space areas;
- sensor or sensor cluster specific features:
  - Camera feature interface (8.3);
  - Ultrasonic feature interface (8.4);
- sensor or sensor cluster specific detections:
  - Radar detection interface (9.3);
  - Lidar detection interface (9.4);
  - Camera detection interface (9.5);
  - Ultrasonic detection interface (9.6);
- supportive information:
  - Sensor performance interface (10.3): with specific sensor-attributes for, for example, current field-of-view (FOV) segments, object recognition rates, reference target recognition rates;

- Sensor health information interface (10.4): with specific attributes for, for example, generic sensor statuses, calibration information, sensor cluster definition;
- sensor or sensor cluster specific input:
  - Common sensor input interface (11.3).

A sensor as well as a sensor cluster can provide several logical interfaces. They can provide none, one or several interfaces on object level. Additionally, a sensor as well as a sensor cluster can provide a sensor technology specific interface on feature- as well as detection level. Each sensor as well as each sensor cluster shall provide at least one interface either on object-, feature- or detection level to fulfil this document. Supportive sensor interfaces can be provided optionally. Sensors and sensor clusters may serve interfaces on all levels. Sensors mainly serve detection level interfaces (DLIs) whereas sensor clusters mainly serve feature level interfaces (FLIs) and/or object level interfaces (OLIs). Sensor clusters are based on sensors and therefore, sensor clusters without a group of defined sensors should be avoided. If a sensor cluster offers no DLIs, the sensor cluster can be considered as a single sensor with multiple sensing elements. A sensor cluster provides consistent data over all its interfaces. The fusion unit does not have to use each provided interface of the sensor or sensor cluster. The provided interfaces can be used by other applications (for example, AD functions), not only by the fusion unit alone.



**Figure 10 —Example of a camera sensor cluster and its sensor and sensor cluster interfaces**

In Figure 10, sensor 1, sensor 2, sensor 3, sensor 4 and sensor sub-cluster {1, 2} and {3, 4} belong to the sensor cluster {1, 2, 3, 4}. The information of all sensors and sensor clusters belongs together and may be referenced between the interfaces. In this example, different combinations of sensors provide interfaces:


- the sensor cluster with the list of the signal “Sensor ID” (A.1.4) values {1, 2, 3, 4} provides the static object interface (SOI), sensor performance interface (SPI) and sensor health information interface (SHII);
- the sensor sub-cluster with the list of the signal “Sensor ID” (A.1.4) values {1, 2} provides potentially the moving object interface (PMOI), camera feature interface (CFI);

- the sensor sub-cluster with the list of the signal “Sensor ID” (A.1.4) values {3, 4} provides PMOI, CFI;
- the sensors with signal “Sensor ID” (A.1.4) value 1, 2, 3 or 4 provide each a camera detection interface (CDI) and SPI;
- the sensor or sensor cluster provides no free space area object interface;
- the sensor or sensor cluster may not receive a sensor input interface.

## 6.2 Generic interface header

Table 5 defines the structure of (6.2) as a specific signal grouping. This header is used by each interface of this document and is adapted or extended accordingly. However, all the interface headers have at least these or similar signals.

**Table 5 — Specific signal grouping: Generic interface header**

| Signal   | RL signal | Option  |
|--|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1)   | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)   | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)  | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|  Sensor ID (A.1.4) | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
| Time stamp – <...> (A.1.5)   | Mandatory |   |
| <...> counter (A.1.6)  | Optional  | Redundancy (B.1.2)<br>Signal: Time stamp – <...> (A.1.5)  |
| Interface cycle time (A.1.7)   | Optional  |   |
| Interface cycle time – variation (A.1.8)   | Optional  |   |
| Data qualifier (A.1.9)   | Mandatory | Default enumerators (B.1.7)   |
| Interface specific header extensions   |           |   |
| Recognised <...> – capability (A.1.10)   | Optional  |   |
| Recognised <...> – status (A.1.11)   | Optional  | Redundancy (B.1.2)<br>Signal: Recognised <...> – capability (A.1.10)<br>Default enumerators (B.1.7)                           |
| Number of valid <...> (A.1.12)   | Mandatory | Optimise LL (B.1.1)   |

## 6.3 Generic interface entity

The interface entities define the set of signals without the header signals of the interface. Each logical interface provides at least one list of interface specific entities (except SHII and CSII). The number of valid entities of each interface entity type list is defined by the signal “Number of valid <...>” (A.1.12) (see Table 5). Interfaces with multiple entity types may not implement all interface entity type lists [see (B.2.3) for conditional entity types]. The signals “Recognised <...> – capability” (A.1.10) and “Recognised

<...> – status” (A.1.11) determine the minimal performance of the sensor for the interface entity type list, for example, at full load.

## 6.4 Profile: Uniqueness of interface versioning

The concatenation of the signal’s “Interface version ID {major, minor, patch}” (A.1.1) value, the signal’s “Interface ID” (A.1.2) value and the signal’s “Sensor ID” (A.1.4) values of a provided interface is a unique identifier for the interpretation of the interface message implementation. The signal “Number of valid serving sensors” (A.1.3) is also included. The order of the signal’s “Sensor ID” (A.1.4) values is also relevant. If a sensor could provide an interface more than once, this profile is adjusted in the corresponding interface to distinguish each interface implementation of one interface.

Sensor input interfaces define a profile for uniqueness of interface versioning (11.2.3).

## 7 Object level

### 7.1 General

The object level interfaces (OLI) originate from several sensors. Therefore, at object level, the term sensor cluster is always used, even if a single sensor is serving the interface. The OLIs provide recognised real-world entities and recognition is based on historical data. Recognised objects are tracked and filtered over time. Model-based algorithms classify these objects and rate them with an existence probability value. The sensor cluster refers its recognised objects to one of two predefined vehicle coordinate systems [see signal “Vehicle coordinate system type – header” (A.1.20)]. The properties, and therefore the signals of the recognised objects, are generally independent of the sensor cluster’s sensor technology. Additional properties of the recognised objects can depend on the sensor technology and complement the object description of the interface. These sensor technology dependent signals are grouped in LSGs for each sensor technology.

To link recognised objects with non-OLI interface entities of the sensor cluster or its sensors, each recognised object has a unique signal’s “Object ID” (A.2.2) value that does not change over time and which is referenced by these entities. The signal’s “Object ID” (A.2.2) value can only be reused after the object has not been recognised for a defined amount of time and the sensor cluster and the fusion unit are therefore certain that the object is no longer visible. Specifically, a reidentification of a previous recognised object with the same signal’s “Object ID” (A.2.2) value, after it has not been visible to the sensor cluster for some time, is not guaranteed.

To link recognised objects of the sensor cluster’s OLIs, the set of related objects use the same consistent unique signal’s “Object grouping ID” (A.2.3) value to link all related objects. One dedicated value may indicate that the object is not linked to another object.

The object recognition algorithm is based on multiple measurements of the sensor cluster and is updated over time. For this reason, the entire object properties are not assigned to the time of one single measurement, but to an estimated point in time at which the properties of the interface’s objects fit together consistently. The object existence probability relates to this consistent set of estimated signals (tracked or predicted quantity values) of the object at this estimated point in time.

On object level the following interfaces are available:

- Potentially moving object interface (7.3);
- Road object interface (7.4);
- Static object interface (7.5);
- Free space area object interface (7.6).

## 7.2 Generic object level interface

Table 6 defines the generic structure of an OLI.

**Table 6 — Generic object level interface**


| Structure  | Multiplicity | Option                      |
|--|--------------|-----------------------------|
| Generic object level header (7.2.1)  | 1            |                             |
| Generic object level entity (7.2.2), specifically an individual object level entity list | Multiple     | Size type:<br>dynamic/fixed |

Each OLI's individual object level entity list can also be a set of different, specific object type lists [for example, for subclause "Road object interface" (7.4) there are two entity lists for the object types: road marking and road boundary and a single entity for the object type: road surface].

### 7.2.1 Generic object level header

Table 7 defines the interface header in subclause "Generic object level header" (7.2.1) and the changes due to the adaptation in comparison to the generic interface header in subclause "Generic interface header" (6.2).

**Table 7 — Specific signal grouping: Generic object level header**

| Signal   | RL signal | Option  |
|--|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1)   | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)   | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)  | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTP (B.1.6)<br>Key: Sensor ID (A.1.4) |
|  Sensor ID (A.1.4) | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
| Time stamp – prediction (A.1.5.1)  | Mandatory |   |
| Cycle counter (A.1.6.1)  | Optional  | Redundancy (B.1.2)<br>Signal: Time stamp – prediction (A.1.5.1)   |
| Interface cycle time (A.1.7)   | Optional  |   |
| Interface cycle time – variation (A.1.8)   | Optional  |   |
| Data qualifier (A.1.9)   | Mandatory | Default enumerators (B.1.7)   |
| OLI specific header extensions   |           |   |
| Recognised <...> – capability (A.1.10)   | Optional  |   |
| Recognised <...> – status (A.1.11)   | Optional  | Redundancy (B.1.2)<br>Signal: Recognised <...> – capability (A.1.10)<br>Default enumerators (B.1.7)                           |
| Number of valid <...> (A.1.12)   | Mandatory | Optimise LL (B.1.1)   |

The following LSGs, which normally shall be provided by the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor cluster (see B.3.3):

- Interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- Interface: SHII; LSG “Calibration” (see Table 53);
- Interface: SHII; LSG “Sensor cluster” (see Table 53).

### 7.2.2 Generic object level entity

Table 8 defines the generic signal grouping “Generic object level entity status” for each OLI object type, except of road surface.

**Table 8 — Specific signal grouping: Generic object level entity status**

| Signal   | RL signal | Option   |
|--|-----------|--|
| Existence probability – object level (A.2.1)                         | Mandatory |  |
| Object ID (A.2.2)  | Mandatory | Alternative A2I (B.3.2)  |
| Object grouping ID (A.2.3)   | Optional  |  |
| Age (A.2.4)  | Mandatory |  |
| Number of valid observations – object level (A.2.5)                  | Optional  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
| <input type="checkbox"/> Time stamp reference – object level (A.2.6) | Mandatory | Alternative VRO (B.1.3)  |
| <input type="checkbox"/> Observation status – object level (A.2.7)   | Mandatory | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
| Track quality (A.2.8)  | Optional  |  |
| Measurement status – object level (A.2.9)                            | Mandatory | Default enumerators (B.1.7)  |

Each interface type adds individual properties to the interface entity types. Geometric information of the object references the vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)]. Detections and features of the sensor cluster and its sensors shall reference objects of the OLIs with the signal “Object ID” (A.2.2). Potentially connected objects of a sensor cluster shall be linked together, and the sensor cluster shall use the same signal’s “Object grouping ID” (A.2.3) value for all objects of this connected object group.

### 7.3 Potentially moving object interface


At object level, the potentially moving object interface (PMOI) contains the list of objects that could potentially move and are relevant for driving situations. This includes all objects that currently move, have moved (see Figure 11) and that are not definitely static three-dimensional structures. Table 9 defines the logical structure of the potentially moving object interface. The object interface defines sets of signals for more detailed potentially moving object types (for example, lights, persons).





Table 9 — Potentially moving object interface

| LSG                       | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option  |
|---------------------------|-----------------|---|--------------------|---|
| Information:<br>interface | M               | Interface version ID {major, minor, patch} (A.1.1)                          | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)  |
|                           |                 | Interface ID (A.1.2)  | O                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Default<br>enumerators<br>(B.1.7)   |
|                           |                 | Number of valid serving sensors<br>(A.1.3)                                  | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Optimise LL<br>(B.1.1)<br>Alternative UTL<br>(B.1.6)<br>Key: Sensor ID<br>(A.1.4) |
|                           |                 | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors (A.1.3) |                    |   |
|                           |                 | Sensor ID (A.1.4)   | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Alternative VRO<br>(B.1.3)  |
|                           |                 | Time stamp – prediction (A.1.5.1)   | M                  |   |
|                           |                 | Cycle counter (A.1.6.1)   | O                  | Redundancy<br>(B.1.2)<br>Signal: Time<br>stamp –<br>prediction<br>(A.1.5.1)   |
|                           |                 | Interface cycle time (A.1.7)  | O                  |   |
|                           |                 | Interface cycle time – variation<br>(A.1.8)                                 | O                  |   |
|                           |                 | Data qualifier (A.1.9)  | M                  | Default<br>enumerators<br>(B.1.7)   |
|                           |                 | Motion type (A.1.13)  | M                  | Profile: Motion<br>type (7.3.3)<br>Default<br>enumerators<br>(B.1.7)  |



| LSG  | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|--|-----------------|---|--------------------|--|
| <b>Potentially moving objects</b>  | M               | Recognised potentially moving objects – capability (A.1.10.1)                             | 0                  |  |
|  |                 | Recognised potentially moving objects – status (A.1.11.1)                                 | 0                  | Redundancy (B.1.2)<br>Signal: Recognised potentially moving objects – capability (A.1.10.1)<br>Default enumerators (B.1.7)   |
|  |                 | Number of valid potentially moving objects (A.1.12.1)                                     | M                  | Optimise LL (B.1.1)  |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid potentially moving objects (A.1.12.1) |                    |  |
|  Potentially moving object status | M               | Existence probability – object level (A.2.1)  | M                  |  |
|  |                 | Object ID (A.2.2)   | M                  | Alternative A2I (B.3.2)  |
|  |                 | Object grouping ID (A.2.3)  | 0                  |  |
|  |                 | Age (A.2.4)   | M                  |  |
|  |                 | Number of valid observations – object level (A.2.5)                                       | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5)   |                    |  |
|  |                 | Time stamp reference – object level (A.2.6)   | M                  | Alternative VRO (B.1.3)  |

| LSG  | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O | Option  |
|--|-----------------|--|--------------------|---|
|  |                 | Observation status – object level (A.2.7)  | M                  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7) |
|  |                 | Track quality (A.2.8)  | O                  |   |
|  |                 | Measurement status – object level (A.2.9)  | M                  | Default enumerators (B.1.7)   |
| <input type="checkbox"/> Potentially moving object information | M               | Tracking motion model (A.2.10)   | O                  | Default enumerators (B.1.7)   |
|  |                 | Number of valid potentially moving object classifications (A.2.11)                                     | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Potentially moving object classification type (A.2.12)     |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid potentially moving object classifications (A.2.11) |                    |   |
|  |                 | Potentially moving object classification type (A.2.12)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|  |                 | Potentially moving object classification type – confidence (A.2.13)                                    | M                  |   |
| <input type="checkbox"/> Potentially moving object position    | M               | Reference point (A.2.14)   | O                  | Default enumerators (B.1.7)   |
|  |                 | Position – object level {x, y, z} (A.2.15)   | {M, M, O}          | Profile: Motion state vector (7.3.4)  |
|  |                 | Position – object level {x, y, z} – error (A.2.16)   | {M, M, O}          | Profile: Motion state vector (7.3.4)<br>Implementation EM (B.4.1)   |

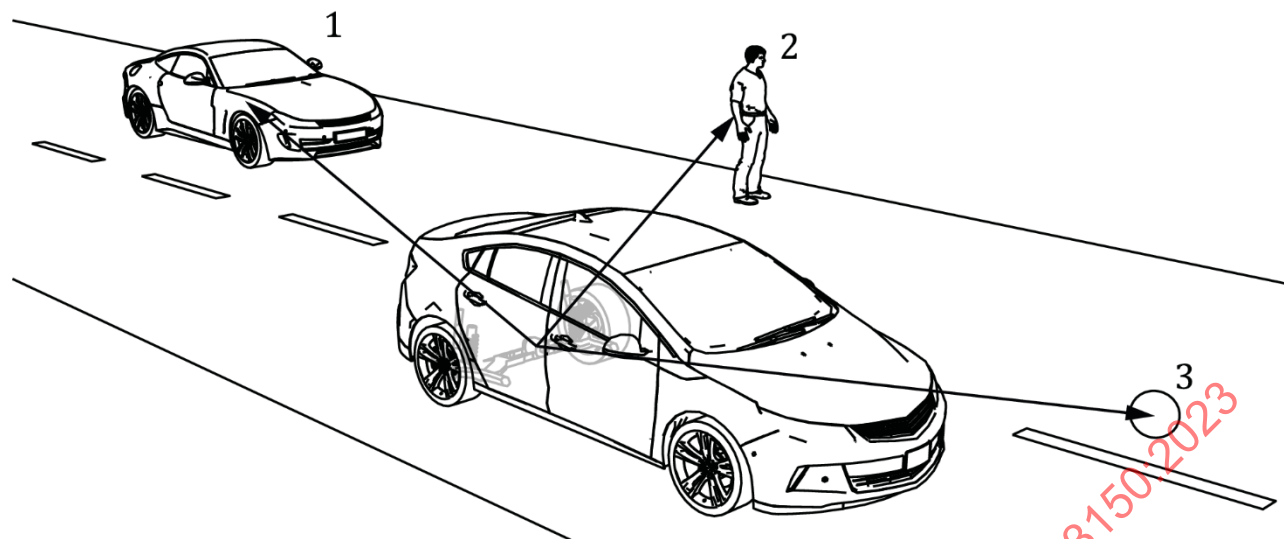
| LSG  | RL LSG<br>M/C/O               | Signal   | RL signal<br>M/C/O  | Option   |
|--|-------------------------------|--|---|--|
|  |                               | Orientation {yaw, pitch, roll} (A.2.17)                      | {C (B.2.1)<br>Relevant: camera, 0, 0}   | Profile: Motion state vector (7.3.4)   |
|  |                               | Orientation {yaw, pitch, roll} – error (A.2.18)              | 0   | Profile: Motion state vector (7.3.4)<br>Implementation EM (B.4.1)  |
|  |                               | Road level (A.2.19)  | 0   | Redundancy (B.1.2)<br>Signal: {z} of Position – object level {x, y, z} (A.2.15)<br>Default enumerators (B.1.7) |
|  Potentially moving object bounding box | C (B.2.1)<br>Relevant: camera | Bounding box extent {length, width, height} (A.2.20)         | {M, M, 0}   |  |
|  |                               | Bounding box extent {length, width, height} – error (A.2.21) | 0   | Implementation EM (B.4.1)  |
|  |                               | Clearance of the object {height} (A.2.22)                    | 0   |  |
|  |                               | Included geometric structures (A.2.23)                       | 0   | Default enumerators (B.1.7)  |
|  Potentially moving object dynamics   | M                             | Velocity {x, y, z} – object level (A.2.24)                   | {C, C, C (B.2.2)<br>Signal: Position {z} (A.2.15)} (B.2.1)<br>Relevant: except ultrasonic | Profile: Motion type (7.3.3)<br>Profile: Motion state vector (7.3.4)   |
|  |                               | Velocity {x, y, z} – object level – error (A.2.25)           | 0   | Profile: Motion type (7.3.3)<br>Profile: Motion state vector (7.3.4)<br>Implementation EM (B.4.1)              |
|  |                               | Acceleration {x, y, z} (A.2.26)                              | 0   | Profile: Motion type (7.3.3)   |
|  |                               | Acceleration {x, y, z} – error (A.2.27)                      | 0   | Profile: Motion type (7.3.3)<br>Implementation EM (B.4.1)  |

| LSG   | RL LSG<br>M/C/O               | Signal   | RL signal<br>M/C/O  | Option  |
|---|-------------------------------|--|---|---|
|   |                               | Instantaneous centre of rotation {x, y} (A.2.28)                         | C (B.2.2)<br>Signal: Rotation rate at instantaneous centre of rotation {yaw} (A.2.30) |   |
|   |                               | Instantaneous centre of rotation {x, y} – error (A.2.29)                 | 0   | Implementation EM (B.4.1)   |
|   |                               | Rotation rate at instantaneous centre of rotation {yaw} (A.2.30)         | 0   | Profile: Motion type (7.3.3)<br>Profile: Motion state vector (7.3.4)                              |
|   |                               | Rotation rate at instantaneous centre of rotation {yaw} – error (A.2.31) | 0   | Profile: Motion type (7.3.3)<br>Profile: Motion state vector (7.3.4)<br>Implementation EM (B.4.1) |
|   |                               | Movement status (A.2.32)   | 0   | Default enumerators (B.1.7)   |
| <input type="checkbox"/> Potentially moving object's lights | C (B.2.1)<br>Relevant: camera | Number of valid lights (A.2.33)  | M   | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Light type (A.2.34)                        |
|   |                               | Size type: dynamic/fixed<br>Size #: Number of valid lights (A.2.33)      |   |   |
|   |                               | Light type (A.2.34)  | M   | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|   |                               | Light status (A.2.35)  | M   | Default enumerators (B.1.7)   |
| <input type="checkbox"/> Potentially moving object: persons | 0                             | Number of valid persons (A.2.36)   | M   | Optimise LL (B.1.1)   |
|   |                               | Size type: dynamic/fixed<br>Size #: Number of valid persons (A.2.36)     |   |   |
|   |                               | Number of valid person type roles (A.2.37)                               | M   | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Person type role (A.2.38)                  |

| LSG | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|-----|-----------------|---|--------------------|--|
|     |                 | Size type: dynamic/fixed<br>Size #: Number of valid person type roles (A.2.37)        |                    |  |
|     |                 | Person type role (A.2.38)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)   |
|     |                 | Person type role – confidence (A.2.39)  | M                  |  |
|     |                 | Number of valid person gestures (A.2.40)  | O                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Person gesture indication type (A.2.41) |
|     |                 | Size type: dynamic/fixed<br>Size #: Number of valid person gestures (A.2.40)          |                    |  |
|     |                 | Person gesture indication type (A.2.41)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)   |
|     |                 | Person gesture indication type – confidence (A.2.42)                                  | O                  |  |
|     |                 | Person gesture direction type (A.2.43)  | O                  | Default enumerators (B.1.7)  |
|     |                 | Person gesture direction type – confidence (A.2.44)                                   | O                  |  |
|     |                 | Number of valid person's body part poses (A.2.45)                                     | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Person body part (A.2.46)               |
|     |                 | Size type: dynamic/fixed<br>Size #: Number of valid person's body part poses (A.2.45) |                    |  |
|     |                 | Person body part (A.2.46)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)   |

| LSG | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|-----|-----------------|---|--------------------|--|
|     |                 | Person body part orientation {yaw, pitch, roll} (A.2.47)                        | M                  |  |
|     |                 | Person body part orientation {yaw, pitch, roll} – error (A.2.48)                | 0                  | Implementation EM (B.4.1)  |
|     |                 | Person body part origin {x, y, z} (A.2.49)                                      | 0                  |  |
|     |                 | Person body part origin {x, y, z} – error (A.2.50)                              | 0                  | Implementation EM (B.4.1)  |
|     |                 | Eye state (A.2.51)  | 0                  | Default enumerators (B.1.7)  |
|     |                 | Eye state – confidence (A.2.52)   | 0                  |  |
|     |                 | Viewing fixation time (A.2.53)  | 0                  |  |
|     |                 | Viewing fixation time – error (A.2.54)  | 0                  | Implementation EM (B.4.1)  |
|     |                 | Viewing fixation direction (A.2.55)   | 0                  | Default enumerators (B.1.7)  |
|     |                 | Viewing fixation direction – confidence (A.2.56)                                | 0                  |  |
|     |                 | Action type (A.2.57)  | 0                  | Default enumerators (B.1.7)  |
|     |                 | Action type – confidence (A.2.58)   | 0                  |  |
|     |                 | Number of valid person accessories (A.2.59)                                     | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Person accessory classification type (A.2.60) |
|     |                 | Size type: dynamic/fixed<br>Size #: Number of valid person accessories (A.2.59) |                    |  |
|     |                 | Person accessory classification type (A.2.60)                                   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)   |

| LSG  | RL LSG<br>M/C/O                  | Signal |  |  | RL signal<br>M/C/O | Option                            |
|--|----------------------------------|--------|--|--|--------------------|-----------------------------------|
|  |                                  |        |  | Person accessory<br>classification type –<br>confidence (A.2.61)                                       | 0                  |                                   |
|  |                                  |        |  | Accessory connection to<br>person type (A.2.62)  | 0                  | Default<br>enumerators<br>(B.1.7) |
|  |                                  |        |  | Included geometric<br>structure of accessory<br>(A.2.63)   | 0                  | Default<br>enumerators<br>(B.1.7) |
| ☐ Lane<br>related<br>information             | 0                                |        |  | Object lane association (A.2.64)   | M                  | Default<br>enumerators<br>(B.1.7) |
|  |                                  |        |  | Angle between object edge and<br>lane {left edge right lane, right<br>edge left lane} (A.2.65)         | 0                  |                                   |
|  |                                  |        |  | Angle between object edge and<br>lane {left edge right lane, right<br>edge left lane} – error (A.2.66) | 0                  | Implementation<br>EM (B.4.1)      |
|  |                                  |        |  | Percentage side lane {left,<br>right} (A.2.67)   | 0                  |                                   |
| ☐ Motion<br>related<br>information           | 0                                |        |  | Angular position {azimuth}<br>(A.2.68)   | M                  |                                   |
|  |                                  |        |  | Angular velocity {azimuth}<br>(A.2.69)   | M                  |                                   |
| ☐ Camera<br>sensor<br>technology<br>specific | C (B.2.1)<br>Relevant:<br>camera |        |  | Scale change – object level<br>(A.2.70)  | M                  |                                   |
| ☐ Radar<br>sensor<br>technology<br>specific  | C (B.2.1)<br>Relevant:<br>radar  |        |  | Entity radar cross section<br>(A.2.71)   | M                  |                                   |
| ☐ Lidar<br>sensor<br>technology<br>specific  | C (B.2.1)<br>Relevant:<br>lidar  |        |  | Entity lidar reflectivity (A.2.72)   | M                  |                                   |


**Key**

- 1 vehicle
- 2 pedestrian
- 3 potentially moving object – classification unknown, for example, “Potentially moving object classification type” (A.2.12) with default enumerator “PMOCT\_Unknown”

**Figure 11 — Example for potentially moving objects****7.3.1 Potentially moving object header**

Table 10 defines the interface header in subclause “Potentially moving object header” (7.3.1) and the changes due to the adaptation in comparison to the generic interface header in 7.2.1. The header of the PMOI can contain a list of valid potentially moving object entities (see 7.3.2)

**Table 10 — Specific signal grouping: Potentially moving object header**

| Signal   | RL signal | Option  |
|--|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1)   | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)   | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)  | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|  Sensor ID (A.1.4) | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
| Time stamp – prediction (A.1.5.1)  | Mandatory |   |
| Cycle counter (A.1.6.1)  | Optional  | Redundancy (B.1.2)<br>Signal: Time stamp – prediction (A.1.5.1)   |
| Interface cycle time (A.1.7)   | Optional  |   |
| Interface cycle time – variation (A.1.8)   | Optional  |   |
| Data qualifier (A.1.9)   | Mandatory | Default enumerators (B.1.7)   |



| Signal  | RL signal | Option   |
|---|-----------|--|
| Motion type (A.1.13)  | Mandatory | Profile: Motion type (7.3.3)<br>Default enumerators (B.1.7)  |
| Recognised potentially moving objects – capability (A.1.10.1) | Optional  |  |
| Recognised potentially moving objects – status (A.1.11.1)     | Optional  | Redundancy (B.1.2)<br>Signal: Recognised potentially moving objects – capability (A.1.10.1)<br>Default enumerators (B.1.7) |
| Number of valid potentially moving objects (A.1.12.1)         | Mandatory | Optimise LL (B.1.1)  |

The following LSGs, which normally shall be provided by the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor cluster (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

### 7.3.2 Potentially moving object entity

Each potentially moving object describes a recognised real-world object and consists of several LSGs. Each object, which may move, even if it might have not moved yet since first recognition of the real-world object, is an entity of potentially moving objects.

A potentially moving object is described by the following LSGs for potentially moving objects with attributes for different parts (for example, vehicle, pedestrian).

- **Status:** the status describes general information of the tracked potentially moving object. This information is based on history of the object tracking. Table 11 defines the signal grouping “Potentially moving object entity status”. It defines this status LSG of the potentially moving object and redefines the signal grouping “Generic object level entity status” (see Table 8).
- **Information:** the information describes properties such as the confidence of an object type.
- **Position:** the position describes the geometric position of the object relative to the vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)].
- **Bounding box:** the object bounding box describes the 3D rectangular hull that encloses the recognised object.
- **Dynamics:** the dynamics describe the motion information of the tracked potentially moving object with respect to the motion type defined by the signal “Motion type” (A.1.13) [see Profile: Motion type (7.3.3)].
- **Lights:** the lights describe the status of the tracked potentially moving object’s lights.
- **Persons:** the persons describe recognised person type, its gestures and poses of the tracked persons if the tracked potentially moving object may be a pedestrian or include persons (for example, the driver of a vehicle).

- **Lane related information:** the lane related information describes the tracked potentially moving object in relation with real-world lanes.
- **Motion related information:** the motion related information describes additional motion information of the tracked potentially moving object.
- **Camera sensor technology specific:** the camera sensor technology specific data describe dynamic information in the camera projection plane.
- **Radar sensor technology specific:** the radar sensor technology specific data describe different surface and material properties of the potentially moving object with respect to the radar.
- **Lidar sensor technology specific:** the lidar sensor technology specific data describes different surface and material properties of the potentially moving object with respect to the lidar (for example, reflection strength of a surface).

**Table 11 — Specific signal grouping: Potentially moving object entity status**

| Signal   | RL signal | Option   |
|--|-----------|--|
| Existence probability – object level (A.2.1)                         | Mandatory |  |
| Object ID (A.2.2)  | Mandatory | Alternative A2I (B.3.2)  |
| Object grouping ID (A.2.3)   | Optional  |  |
| Age (A.2.4)  | Mandatory |  |
| Number of valid observations – object level (A.2.5)                  | Optional  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
| <input type="checkbox"/> Time stamp reference – object level (A.2.6) | Mandatory | Alternative VRO (B.1.3)  |
| <input type="checkbox"/> Observation status – object level (A.2.7)   | Mandatory | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
| Track quality (A.2.8)  | Optional  |  |
| Measurement status – object level (A.2.9)                            | Mandatory | Default enumerators (B.1.7)  |

Potentially moving objects or other object entities on object level may be linked together (for example, a truck with a trailer or a construction site vehicle with a traffic sign at the rear of the vehicle) by using the same signal's “Object grouping ID” (A.2.3) value in all connected objects.

### 7.3.3 Profile: Motion type

The signal “Motion type” (A.1.13) defines the interpretation of the signal “Velocity {x, y, z} – object level” (A.2.24), “Velocity {x, y, z} – object level – error” (A.2.25), “Acceleration {x, y, z}” (A.2.26), “Acceleration {x, y, z} – error” (A.2.27), “Rotation rate at instantaneous centre of rotation {yaw}” (A.2.30) and “Rotation rate at instantaneous centre of rotation {yaw} – error” (A.2.31) values.

### 7.3.4 Profile: Motion state vector

A potentially moving object entity provides a motion state vector. The sensor may provide a full or enhanced error model according to B.4.1.3.5. For example, the motion state vector of the sensor is provided together with an enhanced error model {“Position – object level {x, y, z}” (A.2.15), “Velocity {x, y, z} – object level” (A.2.24), “Orientation {yaw, pitch, roll}” (A.2.17) and “Rotation rate at instantaneous centre of rotation {yaw}” (A.2.30)}.

The resulting error model based on variances, covariances and cross-covariances (see B.4.1.3.4 and B.4.1.3.5) may provide a symmetric nD matrix:

$$\begin{pmatrix} \text{Position – error (A. 2.16)} & \text{Position x Velocity – error} & \text{Position x Orientation – error} & \text{Position x Rotation rate – error} \\ \text{Velocity x Position – error} & \text{Velocity – error (A. 2.25)} & \text{Velocity x Orientation – error} & \text{Velocity x Rotation rate – error} \\ \text{Orientation x Position – error} & \text{Orientation x Velocity – error} & \text{Orientation – error (A. 2.17)} & \text{Orientation x Rotation rate – error} \\ \text{Rotation rate x Position – error} & \text{Rotation rate x Velocity – error} & \text{Rotation rate x Orientation – error} & \text{Rotation rate – error (A. 2.30)} \end{pmatrix}$$

In addition to existing error signals “Position – object level {x, y, z} – error” (A.2.16), “Velocity {x, y, z} – object level – error” (A.2.25), “Orientation {yaw, pitch, roll} – error” (A.2.18) and “Rotation rate at instantaneous centre of rotation {yaw} – error” (A.2.31); the additional signals may be provided as sets of cross-covariances. All covariances of the motion state vector error model are symmetric and therefore additional optimisation methods are applicable; this means only half of non-diagonal elements need to be provided.

For example, the LSG “Potentially moving object motion state vector cross-covariances” may provide, for example, “Position {x, y, z} x Velocity {x, y, z} – object level – error”, “Position {x, y, z} x Orientation {yaw, pitch, roll} – object level – error”. The redundant symmetrical signals for the cross-covariances: “Velocity {x, y, z} x Position {x, y, z} – object level – error”, “Orientation {yaw, pitch, roll} x Position {x, y, z} – object level – error” may be skipped. Similar, the symmetric covariances of the signals “Position – object level {x, y, z} – error” (A.2.16), “Velocity {x, y, z} – object level – error” (A.2.25), “Orientation {yaw, pitch, roll} – error” (A.2.18) and “Rotation rate at instantaneous centre of rotation {yaw} – error” (A.2.31) may not transmit redundant values (see optimisation method B.1.5).

Table 12 provides a subset of the PMOI with the relevant signals for the motion state vector of this profile and the additional signals in the LSG “Potentially moving object motion state vector cross-covariances”.

**Table 12 — Signal grouping: Exemplary motion state vector cross-covariances LSG**

|                                      |   |  |  |   |
|--------------------------------------|---|--|--|---|
| ☐ Potentially moving object position | M | Position – object level {x, y, z} (A.2.15)         | {M, M, O}                                | Profile: Motion state vector (7.3.4)        |
|                                      |   | Position – object level {x, y, z} – error (A.2.16) | {M, M, O}                                | Profile: Motion state vector (7.3.4)<br>... |
|                                      |   | Orientation {yaw, pitch, roll} (A.2.17)            | {C (B.2.1)<br>Relevant:<br>camera, O, O} | Profile: Motion state vector (7.3.4)        |
|                                      |   | Orientation {yaw, pitch, roll} – error (A.2.18)    | O  | Profile: Motion state vector (7.3.4)<br>... |
|                                      |   | ...  | ...                                      | ...   |

|   |   |   |   |   |
|---|---|---|---|---|
| <div><input type="checkbox"/> Potentially moving object dynamics</div>                              | M | Velocity {x, y, z} – object level (A.2.24)  | {C, C, C (B.2.2)<br>Signal:<br>Position {z} (A.2.15)} (B.2.1)<br>Relevant:<br>except ultrasonic | Profile: Motion state vector (7.3.4)<br>... |
|   |   | Velocity {x, y, z} – object level – error (A.2.25)  | 0   | Profile: Motion state vector (7.3.4)<br>... |
|   |   | Rotation rate at instantaneous centre of rotation {yaw} (A.2.30)  | 0   | Profile: Motion state vector (7.3.4)<br>... |
|   |   | Rotation rate at instantaneous centre of rotation {yaw} – error (A.2.31)  | 0   | Profile: Motion state vector (7.3.4)<br>... |
|   |   | ...   | ...   | ...   |
| <div><input type="checkbox"/> Potentially moving object motion state vector cross-covariances</div> | 0 | Position {x, y, z} x Velocity {x, y, z} – object level – error  | 0 <sup>a</sup>  | Implementation EM (B.4.1)                   |
|   |   | Position {x, y, z} x Orientation {yaw, pitch, roll} – object level – error                                      | 0   | Implementation EM (B.4.1)                   |
|   |   | Position {x, y, z} x Rotation rate at instantaneous centre of rotation {yaw} – object level – error             | 0   | Implementation EM (B.4.1)                   |
|   |   | Velocity {x, y, z} x Orientation {yaw, pitch, roll} – object level – error                                      | 0   | Implementation EM (B.4.1)                   |
|   |   | Velocity {x, y, z} x Rotation rate at instantaneous centre of rotation {yaw} – object level – error             | 0   | Implementation EM (B.4.1)                   |
|   |   | Orientation {yaw, pitch, roll} x Rotation rate at instantaneous centre of rotation {yaw} – object level – error | 0   | Implementation EM (B.4.1)                   |

<sup>a</sup> The signal's RL is optional by definition of B.4.1.3.5 but for this profile (7.3.4) it should be defined as mandatory during the system design phase so that at least one signal is mandatory in the LSG.

Additionally, in most cases the description of the signals in planar space (XY-plane) is sufficient. Therefore, below covariances may either be only 2D (position, velocity) or 1D (orientation, rotation rate).

Optimisation method may be also applied by selection of most relevant covariance matrix elements (or cross-covariance matrix elements).

Missing covariance values shall be substituted by suitable values in the fusion tracking system to guarantee a stabile tracking.


## 7.4 Road object interface

The road object interface (RDOI) consists of a general road surface information and two types of objects: road markings and road boundaries (see Figure 12). Table 13 defines the logical structure of the road object interface.

Table 13 — Road object interface



| LSG                       | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option  |
|---------------------------|-----------------|---|--------------------|---|
| Information:<br>interface | M               | Interface version ID {major, minor, patch} (A.1.1)                          | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)  |
|                           |                 | Interface ID (A.1.2)  | O                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Default<br>enumerators<br>(B.1.7)   |
|                           |                 | Number of valid serving sensors<br>(A.1.3)                                  | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Optimise LL<br>(B.1.1)<br>Alternative UTL<br>(B.1.6)<br>Key: Sensor ID<br>(A.1.4) |
|                           |                 | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors (A.1.3) |                    |   |
|                           |                 | Sensor ID (A.1.4)   | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Alternative VRO<br>(B.1.3)  |
|                           |                 | Time stamp – prediction (A.1.5.1)   | M                  |   |
|                           |                 | Cycle counter (A.1.6.1)   | O                  | Redundancy<br>(B.1.2)<br>Signal: Time stamp<br>– prediction<br>(A.1.5.1)  |
|                           |                 | Interface cycle time (A.1.7)  | O                  |   |
|                           |                 | Interface cycle time – variation<br>(A.1.8)                                 | O                  |   |
|                           |                 | Data qualifier (A.1.9)  | M                  | Default<br>enumerators<br>(B.1.7)   |







| LSG          | RL LSG<br>M/C/O                       | Signal  | RL signal<br>M/C/O               | Option  |
|--------------|---------------------------------------|---|----------------------------------|---|
|              |                                       | Colour model type (A.1.14)  | C (B.2.1)<br>Relevant:<br>camera | Profile: Colour model for RDOI (7.4.3)<br>Alternative VRO (B.1.3)<br>Use enumeration to define colours by defining colour values and the applied colour model for each enumerator.<br>Default enumerators (B.1.7) |
| Road surface | C (B.2.3)<br>multiple<br>entity types | Road type – object level (A.2.73)   | M                                | Default enumerators (B.1.7)   |
|              |                                       | Road type – confidence (A.2.74)   | O                                |   |
|              |                                       | Number of valid road surface classifications (A.2.75)                                     | M                                | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Road surface classification type – object level (A.2.76)   |
|              |                                       | Size type: dynamic/fixed<br>Size #: Number of valid road surface classifications (A.2.75) |                                  |   |
|              |                                       | Road surface classification type – object level (A.2.76)                                  | M                                | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|              |                                       | Road surface classification type – confidence (A.2.77)                                    | M                                |   |
|              |                                       | Road surface roughness – object level (A.2.78)  | O                                |   |
|              |                                       | Road surface texture – object level (A.2.79)  | O                                |   |
|              |                                       | Number of valid road surface condition classifications (A.2.80)                           | O                                | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Road surface condition classification type – object level (A.2.81)   |

| LSG   | RL LSG<br>M/C/O                       | Signal  | RL signal<br>M/C/O | Option  |
|---|---------------------------------------|---|--------------------|---|
|   |                                       | Size type: dynamic/fixed<br>Size #: Number of valid road surface condition classifications (A.2.80) |                    |   |
|   |                                       | Road surface condition classification type – object level (A.2.81)                                  | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|   |                                       | Road surface condition classification type – confidence (A.2.82)                                    | M                  |   |
|   |                                       | Track quality (A.2.8)   | O                  |   |
|   |                                       | Measurement status – object level (A.2.9)   | O                  | Default enumerators (B.1.7)   |
| Road markings   | C (B.2.3)<br>multiple<br>entity types | Recognised road markings – capability (A.1.10.2)  | O                  |   |
|   |                                       | Recognised road markings – status (A.1.11.2)  | O                  | Redundancy (B.1.2)<br>Signal: Recognised road markings – capability (A.1.10.2)<br>Default enumerators (B.1.7) |
|   |                                       | Number of valid road markings (A.1.12.2)  | M                  | Optimise LL (B.1.1)   |
|   |                                       | Size type: dynamic/fixed<br>Size #: Number of valid road markings (A.1.12.2)                        |                    |   |
|  Road marking status | M                                     | Existence probability – object level (A.2.1)  | M                  |   |
|   |                                       | Object ID (A.2.2)   | M                  | Alternative A2I (B.3.2)   |
|   |                                       | Object grouping ID (A.2.3)  | O                  |   |
|   |                                       | Age (A.2.4)   | M                  |   |


| LSG   | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|---|-----------------|---|--------------------|--|
|   |                 | Number of valid observations – object level (A.2.5)                                       | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5)   |                    |  |
|   |                 | Time stamp reference – object level (A.2.6)   | M                  | Alternative VRO (B.1.3)  |
|   |                 | Observation status – object level (A.2.7)   | M                  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
|   |                 | Track quality (A.2.8)   | 0                  |  |
|   |                 | Measurement status – object level (A.2.9)   | M                  | Default enumerators (B.1.7)  |
| <input type="checkbox"/> Road marking information | M               | Number of valid road marking classifications (A.2.83)                                     | M                  | Optimise LL (B.1.1)  |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid road marking classifications (A.2.83) |                    |  |
|   |                 | Road marking type (A.2.84)  | M                  | Default enumerators (B.1.7)  |
|   |                 | Road marking type – confidence (A.2.85)   | M                  |  |
|   |                 | Road object lane association (A.2.86)   | 0                  | Default enumerators (B.1.7)  |
|   |                 | Road object lane association – confidence (A.2.87)  | 0                  |  |
|   |                 | Arrow orientation {yaw} (A.2.88)  | 0                  |  |





| LSG   | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O   | Option   |
|---|-----------------|---|--|--|
|   |                 | Arrow direction (A.2.89)  | C (B.2.2)<br>* The signal "Road marking type" (A.2.84) has "RMT_Arrow" or a similar enumerator defined during the system design phase. | Default enumerators (B.1.7)  |
|   Road marking sign information | M               | Number of valid sign classifications (A.2.90)                                     | 0  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sign classification type (A.2.91) |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid sign classifications (A.2.90) |  |  |
|   |                 | Sign classification type (A.2.91)   | M  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)                                   |
|   |                 | Sign classification type – confidence (A.2.92)                                    | M  |  |
|   |                 | Sign value (A.2.93)   | 0  |  |
|   |                 | Sign value unit (A.2.94)  | 0  | Default enumerators (B.1.7)  |
|   |                 | Sign text (A.2.95)  | 0  |  |
|   |                 | Sign text font (A.2.96)   | 0  | Default enumerators (B.1.7)  |
|   |                 | Sign text – confidence (A.2.97)   | 0  |  |
|   |                 | Number of valid entity states (A.2.98)  | 0  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Entity state (A.2.99)             |

| LSG  | RL LSG<br>M/C/O                      | Signal  | RL signal<br>M/C/O | Option   |
|--|--------------------------------------|---|--------------------|--|
|  |                                      | Size type: dynamic/fixed<br>Size #: Number of valid entity states (A.2.98)  |                    |  |
|  |                                      | Entity state (A.2.99)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)                           |
|    Road marking message colour tone | C (B.2.1)<br>Relevant: camera        | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |  |
|  |                                      | Colour value – object level (A.2.100)   | M                  | Profile: Colour model for RDOI (7.4.3)   |
|  |                                      | Colour tone – confidence – object level (A.2.101)   | O                  | Profile: Colour model for RDOI (7.4.3)   |
|   Road marking background colour tone  | C (B.2.1)<br>Relevant: camera        | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |  |
|  |                                      | Colour value – object level (A.2.100)   | M                  | Profile: Colour model for RDOI (7.4.3)   |
|  |                                      | Colour tone – confidence – object level (A.2.101)   | O                  | Profile: Colour model for RDOI (7.4.3)   |
|  Road marking connections   | C (B.2.1)<br>Relevant: camera, lidar | Number of valid connections (A.2.102)   | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Connection type (A.2.103) |
|  |                                      | Size type: dynamic/fixed<br>Size #: Number of valid connections (A.2.102)   |                    |  |
|  |                                      | Connection type (A.2.103)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)                           |
|  |                                      | Connection grouping ID (A.2.104)  | M                  |  |
|  |                                      | Vertex point {x, y, z} (A.2.105)  | {M, M, O}          |  |
|  |                                      | Vertex point {x, y, z} – error (A.2.106)  | {M, M, O}          | Implementation EM (B.4.1)  |

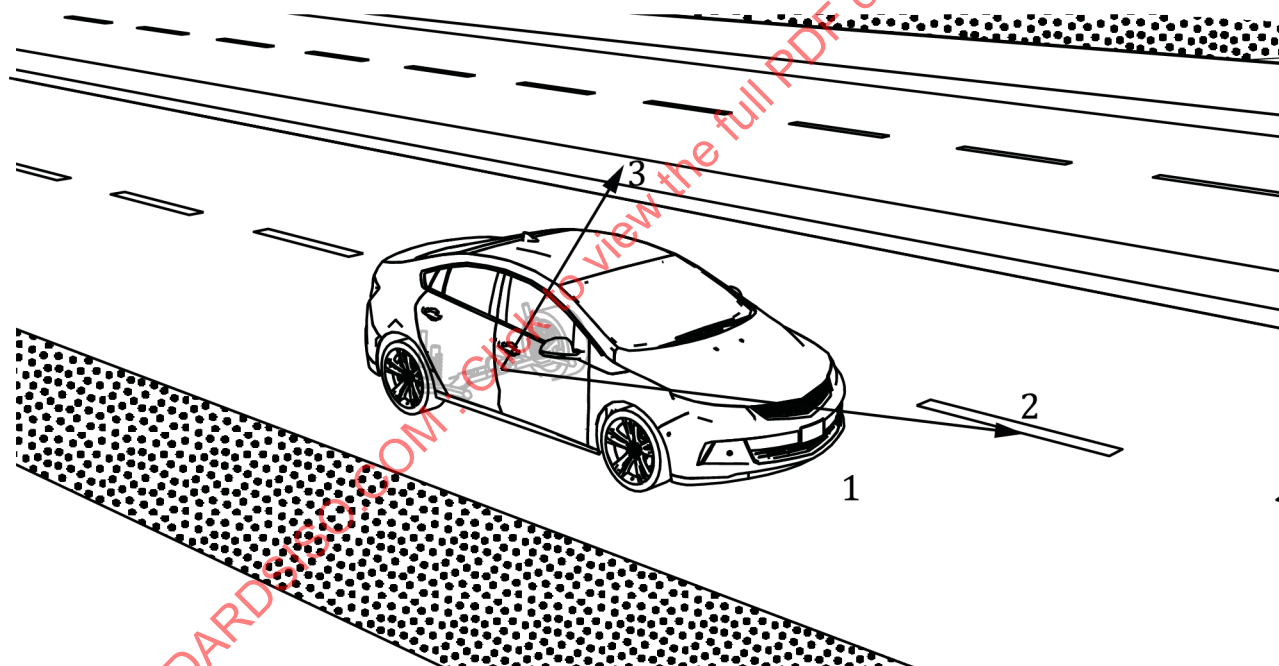
| LSG                        | RL LSG<br>M/C/O                            | Signal  | RL signal<br>M/C/O | Option                      |
|----------------------------|--|---|--------------------|-----------------------------|
| ☐ Road marking polynomials | C (B.2.1)<br>Relevant:<br>camera,<br>lidar | Vertex point – confidence {x, y, z} (A.2.107)                             | 0                  |                             |
|                            |  | Number of valid polynomials (A.2.108)                                     | M                  | Optimise LL (B.1.1)         |
|                            |  | Size type: dynamic/fixed<br>Size #: Number of valid polynomials (A.2.108) |                    |                             |
|                            |  | Polynomial coefficient y {c0, c1, c2, c3} (A.2.109)                       | M                  |                             |
|                            |  | Polynomial coefficient z {c0, c1, c2, c3} (A.2.110)                       | 0                  |                             |
|                            |  | Polynomial y – error (A.2.111)  | 0                  | Implementation EM (B.4.1)   |
|                            |  | Polynomial z – error (A.2.112)  | 0                  | Implementation EM (B.4.1)   |
|                            |  | Polynomial range x {begin, end} (A.2.113)                                 | M                  |                             |
|                            |  | Extent {width, height} – polynomial (A.2.114)                             | 0                  |                             |
|                            |  | Extent {width, height} – polynomial – error (A.2.115)                     | 0                  | Implementation EM (B.4.1)   |
|                            |  | Extent {width, height} – polynomial – confidence (A.2.116)                | 0                  |                             |
|                            |  | Number of valid data ranges (A.2.117)                                     | 0                  | Optimise LL (B.1.1)         |
|                            |  | Size type: dynamic/fixed<br>Size #: Number of valid data ranges (A.2.117) |                    |                             |
|                            |  | Supported data range x {begin, end} (A.2.118)                             | M                  |                             |
|                            |  | Supported axis (A.2.119)  | M                  |                             |
| ☐ Road marking polylines   | C (B.2.1)<br>Relevant:<br>camera,<br>lidar | Polyline interpolation method (A.2.120)                                   | M                  | Default enumerators (B.1.7) |
|                            |  | Number of valid polylines (A.2.121)                                       | M                  | Optimise LL (B.1.1)         |
|                            |  | Size type: dynamic/fixed<br>Size #: Number of valid polylines (A.2.121)   |                    |                             |
|                            |  | Number of valid vertices (A.2.122)  | M                  | Optimise LL (B.1.1)         |
|                            |  | Size type: dynamic/fixed<br>Size #: Number of valid vertices (A.2.122)    |                    |                             |

| LSG  | RL LSG<br>M/C/O                       | Signal   | RL signal<br>M/C/O | Option   |
|--|---------------------------------------|--|--------------------|--|
|  |                                       | Vertex point {x, y, z} (A.2.105)   | {M, M, O}          |  |
|  |                                       | Vertex point {x, y, z} – error (A.2.106)                                       | {M, M, O}          | Implementation EM (B.4.1)  |
|  |                                       | Vertex point – confidence {x, y, z} (A.2.107)                                  | O                  |  |
|  |                                       | Extent {width, height} – vertex (A.2.123)                                      | O                  |  |
|  |                                       | Extent {width, height} – vertex – error (A.2.124)                              | O                  | Implementation EM (B.4.1)  |
|  |                                       | Extent {width, height} – vertex – confidence (A.2.125)                         | O                  |  |
| Road boundaries  | C (B.2.3)<br>multiple<br>entity types | Recognised road boundaries – capability (A.1.10.3)                             | O                  |  |
|  |                                       | Recognised road boundaries – status (A.1.11.3)                                 | O                  | Redundancy (B.1.2)<br>Signal: Recognised road boundaries – capability (A.1.10.3)<br>Default enumerators (B.1.7)  |
|  |                                       | Number of valid road boundaries (A.1.12.3)                                     | M                  | Optimise LL (B.1.1)  |
|  |                                       | Size type: dynamic/fixed<br>Size #: Number of valid road boundaries (A.1.12.3) |                    |  |
|  Road boundary status | M                                     | Existence probability – object level (A.2.1)                                   | M                  |  |
|  |                                       | Object ID (A.2.2)  | M                  | Alternative A2I (B.3.2)  |
|  |                                       | Object grouping ID (A.2.3)   | O                  |  |
|  |                                       | Age (A.2.4)  | M                  |  |
|  |                                       | Number of valid observations – object level (A.2.5)                            | O                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |

| LSG                           | RL LSG<br>M/C/O               | Signal  | RL signal<br>M/C/O | Option  |
|-------------------------------|-------------------------------|---|--------------------|---|
|                               |                               | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5)                                   |                    |   |
|                               |                               | Time stamp reference – object level (A.2.6)   | M                  | Alternative VRO (B.1.3)   |
|                               |                               | Observation status – object level (A.2.7)   | M                  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7) |
|                               |                               | Track quality (A.2.8)   | O                  |   |
|                               |                               | Measurement status – object level (A.2.9)   | M                  | Default enumerators (B.1.7)   |
| ☐ Road boundary information   | M                             | Number of valid road boundary classifications (A.2.126)   | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Road boundary type (A.2.127)                               |
|                               |                               | Size type: dynamic/fixed<br>Size #: Number of valid road boundary classifications (A.2.126)                               |                    |   |
|                               |                               | Road boundary type (A.2.127)  | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|                               |                               | Road boundary type – confidence (A.2.128)   | M                  |   |
|                               |                               | Road object lane association (A.2.86)   | O                  | Default enumerators (B.1.7)   |
|                               |                               | Road object lane association – confidence (A.2.87)  | O                  |   |
| ☐ ☐ Road boundary colour tone | C (B.2.1)<br>Relevant: camera | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |   |
|                               |                               | Colour value – object level (A.2.100)   | M                  | Profile: Colour model for RDOI (7.4.3)  |

| LSG   | RL LSG<br>M/C/O                             | Signal  | RL signal<br>M/C/O | Option                                 |
|---|---|---|--------------------|--|
|   |   | Colour tone – confidence – object level (A.2.101)                         | 0                  | Profile: Colour model for RDOI (7.4.3) |
|  Road boundary polynomials | C (B.2.1)<br>Relevant: camera, lidar, radar | Number of valid polynomials (A.2.108)                                     | M                  | Optimise LL (B.1.1)                    |
|   |   | Size type: dynamic/fixed<br>Size #: Number of valid polynomials (A.2.108) |                    |  |
|   |   | Polynomial coefficient y {c0, c1, c2, c3} (A.2.109)                       | M                  |  |
|   |   | Polynomial coefficient z {c0, c1, c2, c3} (A.2.110)                       | 0                  |  |
|   |   | Polynomial y – error (A.2.111)  | 0                  | Implementation EM (B.4.1)              |
|   |   | Polynomial z – error (A.2.112)  | 0                  | Implementation EM (B.4.1)              |
|   |   | Polynomial range x {begin, end} (A.2.113)                                 | M                  |  |
|   |   | Extent {width, height} – polynomial (A.2.114)                             | 0                  |  |
|   |   | Extent {width, height} – polynomial – error (A.2.115)                     | 0                  | Implementation EM (B.4.1)              |
|   |   | Extent {width, height} – polynomial – confidence (A.2.116)                | 0                  |  |
|   |   | Number of valid data ranges (A.2.117)                                     | 0                  | Optimise LL (B.1.1)                    |
|   |   | Size type: dynamic/fixed<br>Size #: Number of valid data ranges (A.2.117) |                    |  |
|   |   | Supported data range x {begin, end} (A.2.118)                             | M                  |  |
|   |   | Supported axis (A.2.119)  | M                  |  |
|  Road boundary polylines | C (B.2.1)<br>Relevant: camera, lidar, radar | Polyline interpolation method (A.2.120)                                   | M                  | Default enumerators (B.1.7)            |
|   |   | Number of valid polylines (A.2.121)                                       | M                  | Optimise LL (B.1.1)                    |
|   |   | Size type: dynamic/fixed<br>Size #: Number of valid polylines (A.2.121)   |                    |  |
|   |   | Number of valid vertices (A.2.122)  | M                  | Optimise LL (B.1.1)                    |
|   |   | Size type: dynamic/fixed<br>Size #: Number of valid vertices (A.2.122)    |                    |  |

| LSG | RL LSG<br>M/C/O | Signal |  |  | RL signal<br>M/C/O | Option                    |
|-----|-----------------|--------|--|--|--------------------|---------------------------|
|     |                 |        |  | Vertex point {x, y, z} (A.2.105)                       | {M, M, O}          |                           |
|     |                 |        |  | Vertex point {x, y, z} – error (A.2.106)               | {M, M, O}          | Implementation EM (B.4.1) |
|     |                 |        |  | Vertex point – confidence {x, y, z} (A.2.107)          | 0                  |                           |
|     |                 |        |  | Extent {width, height} – vertex (A.2.123)              | 0                  |                           |
|     |                 |        |  | Extent {width, height} – vertex – error (A.2.124)      | 0                  | Implementation EM (B.4.1) |
|     |                 |        |  | Extent {width, height} – vertex – confidence (A.2.125) | 0                  |                           |



#### Key


- 1 road surface; relevant for the ego-vehicle
- 2 road marking
- 3 road boundary

Figure 12 — Example for road objects

#### 7.4.1 Road object header

Table 14 defines the interface header in subclause “Road object header” (7.4.1) and the changes due to the adaptation in comparison to the generic interface header in 7.2.1. The header of the RDOI can contain a list of valid road marking entities, a list of valid road boundary entities and/or a single road surface entity (see 7.4.2).

Table 14 — Specific signal grouping: Road object header

| Signal  | RL signal   |           | Option  |
|---|---|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1)  | Mandatory   |           | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)  | Optional  |           | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)   | Mandatory   |           | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4)   |
|  Sensor ID (A.1.4) | Mandatory   |           | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
| Time stamp – prediction (A.1.5.1)   | Mandatory   |           |   |
| Cycle counter (A.1.6.1)   | Optional  |           | Redundancy (B.1.2)<br>Signal: Time stamp – prediction (A.1.5.1)   |
| Interface cycle time (A.1.7)  | Optional  |           |   |
| Interface cycle time – variation (A.1.8)  | Optional  |           |   |
| Data qualifier (A.1.9)  | Mandatory   |           | Default enumerators (B.1.7)   |
| Colour model type (A.1.14)  | Conditional (B.2.1)<br>Relevant: camera             |           | Profile: Colour model for RDOI (7.4.3)<br>Alternative VRO (B.1.3)<br>Use enumeration to define colours by defining colour values and the applied colour model for each enumerator.<br>Default enumerators (B.1.7) |
| Recognised road markings – capability (A.1.10.2)  | RL LSG conditional (B.2.3)<br>multiple entity types | Optional  |   |
| Recognised road markings – status (A.1.11.2)  |   | Optional  | Redundancy (B.1.2)<br>Signal: Recognised road markings – capability (A.1.10.2)<br>Default enumerators (B.1.7)   |
| Number of valid road markings (A.1.12.2)  |   | Mandatory | Optimise LL (B.1.1)   |
| Recognised road boundaries – capability (A.1.10.3)  |   | Optional  |   |
| Recognised road boundaries – status (A.1.11.3)  |   | Optional  | Redundancy (B.1.2)<br>Signal: Recognised road boundaries – capability (A.1.10.3)<br>Default enumerators (B.1.7)   |
| Number of valid road boundaries (A.1.12.3)  |   | Mandatory | Optimise LL (B.1.1)   |



The following LSGs, which normally shall be provided by the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor cluster (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

#### 7.4.2 Road object entity

Each road object describes a recognised real-world road limitation (road marking or road boundary) or the relevant road surface for the ego-vehicle. They consist of several LSGs.

The road surface is described by the road surface LSG. The road surface provides no LSG “Status”.

A road marking is described by the following LSGs for road marking.

- **Status:** the status describes the general information of a tracked road marking. This information is based on history of the object tracking. Table 15 defines the signal grouping “Road object entity status”. It defines this status LSG of the road marking and redefines the signal grouping “Generic object level entity status” (see Table 8).
- **Information:** the information describes the road marking type and different visual properties related to the road marking.
- **Sign information:** the sign information describes optional the sign symbol of the road marking, which may be equivalent to a traffic sign as traffic main sign.
- **Colour tone:** the colour tone describes the main colour of the message text and the background of the road marking.
- The contour of a road marking could be described by polynomials and/or polylines and connections:
  - **Connections:** the contour can be connected with other road markings contours of the road and each connection has a common vertex point at the connection of the road markings.
  - **Polynomials:** the polynomials of a road marking describe the detailed shape and bounds in a 3D vector space with polynomial lines. Different polynomials describe the y and z values of the contour independently.
  - **Polylines:** the polylines of a road marking describe the detailed shape and bounds in a 3D vector space with multiple line segments and a defined interpolation method.

A road boundary is described by the following LSGs for road boundary:

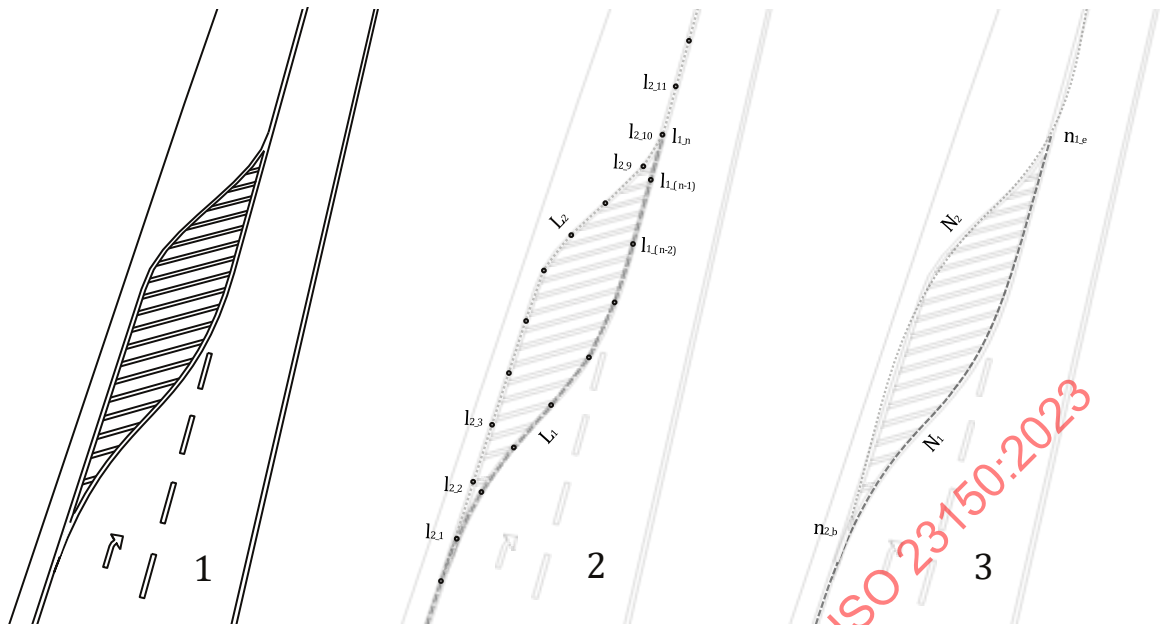
- **Status:** the status describes the general information of a tracked road boundary. This information is based on history of the object tracking. Table 15 defines the signal grouping “Road object entity status”. It defines this status LSG of the road boundary and redefines the signal grouping “Generic object level entity status” (see Table 8).
- **Information:** the information describes the road boundary type and different visual properties related to the road boundary.
- **Colour tone:** the colour tone describes the main colour of the road boundary.

- The contour could be described by polynomials and/or polylines:
  - **Polynomials:** the polynomials of a road boundary describe the detailed shape and bounds in a 3D vector space with polynomial lines. Different polynomials describe the y and z values of the contour independently.
  - **Polylines:** the polylines of a road boundary describe the detailed shape and bounds in a 3D vector space with multiple line segments and a defined interpolation method.

**Table 15 — Specific signal grouping: Road object entity status**

| Signal   | RL signal | Option   |
|--|-----------|--|
| Existence probability – object level (A.2.1)                         | Mandatory |  |
| Object ID (A.2.2)  | Mandatory | Alternative A2I (B.3.2)  |
| Object grouping ID (A.2.3)   | Optional  |  |
| Age (A.2.4)  | Mandatory |  |
| Number of valid observations – object level (A.2.5)                  | Optional  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
| <input type="checkbox"/> Time stamp reference – object level (A.2.6) | Mandatory | Alternative VRO (B.1.3)  |
| <input type="checkbox"/> Observation status – object level (A.2.7)   | Mandatory | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
| Track quality (A.2.8)  | Optional  |  |
| Measurement status – object level (A.2.9)                            | Mandatory | Default enumerators (B.1.7)  |

Road markings and road boundaries can be represented either by polynomials of degree 3 or by polylines consisting of one or more segments merged at the vertex points (see Figure 13 and see informative figures in Annex A for details: Figure A.11 and Figure A.14).

**Key**

1 real-world scene

2 polyline contour representation

 $L_x$   $x^{\text{th}}$  polyline $l_{x,y}$   $y^{\text{th}}$  vertex point [signal “Vertex point {x, y, z}” (A.2.105)] of the  $x^{\text{th}}$  polyline

3 polynomial contour representation

 $N_x$   $x^{\text{th}}$  polynomial $n_{x,b}$  range begin of the  $x^{\text{th}}$  polynomial [see “Polynomial range x {begin, end}” (A.2.113)] $n_{x,e}$  range end of the  $x^{\text{th}}$  polynomial [see “Polynomial range x {begin, end}” (A.2.113)]**Figure 13 — Example for alternative contour representations**

The road marking contours are specified with road marking polynomials and/or road marking polylines and refer to connections of road markings by the signal “Connection type” (A.2.103) and the signal “Connection grouping ID” (A.2.104). Each road marking connection is defined by a unique signal “Connection grouping ID” (A.2.104) value and connects two or more road marking contours.

**7.4.3 Profile: Colour model for RDOI**

The profile is relevant for camera sensor technology (B.2.1). The signal “Colour model type” (A.1.14) defines the colour model for the interface. Depending on the colour model, each colour is described by a fixed number of colour values [see signal “Colour value – object level” (A.2.100)] to define the colour tone. The confidence of the colour tone is provided by the signal “Colour tone – confidence – object level” (A.2.101).

The interface provides colour tones for road markings and road boundaries.


**7.5 Static object interface**

The static object interface (SOI) consists of four types of entities: general landmarks, traffic signs (traffic main sign and additional supplementary signs), traffic sign boards and traffic lights (see Figure 14). Table 16 defines the logical structure of the static object interface.

**Table 16 — Static object interface**

| LSG | RL LSG<br>M/C/O | Signal | RL signal<br>M/C/O | Option |
|-----|-----------------|--------|--------------------|--------|
|-----|-----------------|--------|--------------------|--------|

| LSG                    | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|------------------------|-----------------|---|--------------------|--|
| Information: interface | M               | Interface version ID {major, minor, patch} (A.1.1)                          | M                  | Profile:<br>Uniqueness of interface versioning (6.4)   |
|                        |                 | Interface ID (A.1.2)  | O                  | Profile:<br>Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
|                        |                 | Number of valid serving sensors (A.1.3)                                     | M                  | Profile:<br>Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|                        |                 | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors (A.1.3) |                    |  |
|                        |                 | Sensor ID (A.1.4)   | M                  | Profile:<br>Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
|                        |                 | Time stamp – prediction (A.1.5.1)   | M                  |  |
|                        |                 | Cycle counter (A.1.6.1)   | O                  | Redundancy (B.1.2)<br>Signal: Time stamp – prediction (A.1.5.1)  |
|                        |                 | Interface cycle time (A.1.7)  | O                  |  |
|                        |                 | Interface cycle time – variation (A.1.8)                                    | O                  |  |
|                        |                 | Data qualifier (A.1.9)  | M                  | Default enumerators (B.1.7)  |

| LSG   | RL LSG<br>M/C/O                          | Signal   | RL signal<br>M/C/O               | Option   |
|---|--|--|----------------------------------|--|
|   |  | Colour model type (A.1.14)   | C (B.2.1)<br>Relevant:<br>camera | Profile: Colour model for SOI (7.5.3)<br>Alternative VRO (B.1.3)<br>Use enumeration to define colours by defining colour values and the applied colour model for each enumerator.<br>Default enumerators (B.1.7) |
| <b>General landmarks</b>  | C (B.2.3)<br>multiple<br>entity<br>types | Recognised general landmarks – capability (A.1.10.4)                             | 0                                |  |
|   |  | Recognised general landmarks – status (A.1.11.4)                                 | 0                                | Redundancy (B.1.2)<br>Signal: Recognised general landmarks – capability (A.1.10.4)<br>Default enumerators (B.1.7)  |
|   |  | Number of valid general landmarks (A.1.12.4)                                     | M                                | Optimise LL (B.1.1)  |
|   |  | Size type: dynamic/fixed<br>Size #: Number of valid general landmarks (A.1.12.4) |                                  |  |
|  General landmark status | M  | Existence probability – object level (A.2.1)                                     | M                                |  |
|   |  | Object ID (A.2.2)  | M                                | Alternative A2I (B.3.2)  |
|   |  | Object grouping ID (A.2.3)   | 0                                |  |
|   |  | Age (A.2.4)  | M                                |  |
|   |  | Number of valid observations – object level (A.2.5)                              | 0                                | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)]                   |

| LSG                          | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O                    | Option  |
|------------------------------|-----------------|--|---------------------------------------|---|
|                              |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5)        |                                       |   |
|                              |                 | Time stamp reference – object level (A.2.6)  | M                                     | Alternative VRO (B.1.3)   |
|                              |                 | Observation status – object level (A.2.7)  | M                                     | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7) |
|                              |                 | Track quality (A.2.8)  | O                                     |   |
|                              |                 | Measurement status – object level (A.2.9)  | M                                     | Default enumerators (B.1.7)   |
|                              |                 |  |                                       |   |
| General landmark information | M               | Number of valid general landmark classifications (A.2.129)                                     | M                                     | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: General landmark classification type (A.2.130)             |
|                              |                 | Size type: dynamic/fixed<br>Size #: Number of valid general landmark classifications (A.2.129) |                                       |   |
|                              |                 | General landmark classification type (A.2.130)   | M                                     | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|                              |                 | General landmark classification type – confidence (A.2.131)                                    | M                                     |   |
|                              |                 |  |                                       |   |
| General landmark position    | M               | Reference point (A.2.14)   | O                                     | Default enumerators (B.1.7)   |
|                              |                 | Position – object level {x, y, z} (A.2.15)   | {M, M, O}                             |   |
|                              |                 | Position – object level {x, y, z} – error (A.2.16)   | {M, M, O}                             | Implementation EM (B.4.1)   |
|                              |                 | Orientation {yaw, pitch, roll} (A.2.17)  | {C (B.2.1)<br>Relevant: camera, O, O} |   |

| LSG                             | RL LSG<br>M/C/O                          | Signal  | RL signal<br>M/C/O | Option   |
|---------------------------------|--|---|--------------------|--|
|                                 |  | Orientation {yaw, pitch, roll} – error (A.2.18)   | 0                  | Implementation EM (B.4.1)  |
| ☐ General landmark bounding box | C (B.2.1)<br>Relevant:<br>camera         | Bounding box extent {length, width, height} (A.2.20)                                    | {M, M, O}          |  |
|                                 |  | Bounding box extent {length, width, height} – error (A.2.21)                            | 0                  | Implementation EM (B.4.1)  |
| Traffic signs                   | C (B.2.3)<br>multiple<br>entity<br>types | Recognised traffic signs – capability (A.1.10.5)  | 0                  |  |
|                                 |  | Recognised traffic signs – status (A.1.11.5)  | 0                  | Redundancy (B.1.2)<br>Signal: Recognised traffic signs – capability (A.1.10.5)<br>Default enumerators (B.1.7)  |
|                                 |  | Number of valid traffic signs (A.1.12.5)  | M                  | Optimise LL (B.1.1)  |
|                                 |  | Size type: dynamic/fixed<br>Size #: Number of valid traffic signs (A.1.12.5)            |                    |  |
|                                 |  |   |                    |  |
| ☐ Traffic main sign status      | M  | Existence probability – object level (A.2.1)  | M                  |  |
|                                 |  | Object ID (A.2.2)   | M                  | Alternative A2I (B.3.2)  |
|                                 |  | Object grouping ID (A.2.3)  | O                  |  |
|                                 |  | Age (A.2.4)   | M                  |  |
|                                 |  | Number of valid observations – object level (A.2.5)                                     | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|                                 |  | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5) |                    |  |
|                                 |  | Time stamp reference – object level (A.2.6)   | M                  | Alternative VRO (B.1.3)  |








| LSG  | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option  |
|--|-----------------|---|--------------------|---|
| <input type="checkbox"/> Traffic main sign information |                 | Observation status – object level (A.2.7)   | M                  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7) |
|  |                 | Track quality (A.2.8)   | O                  |   |
|  |                 | Measurement status – object level (A.2.9)   | M                  | Default enumerators (B.1.7)   |
|  | M               | Number of valid sign classifications (A.2.90)                                     | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sign classification type (A.2.91)                          |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid sign classifications (A.2.90) |                    |   |
|  |                 | Sign classification type (A.2.91)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|  |                 | Sign classification type – confidence (A.2.92)                                    | M                  |   |
|  |                 | Sign value (A.2.93)   | M                  |   |
|  |                 | Sign value unit (A.2.94)  | M                  | Default enumerators (B.1.7)   |
|  |                 | Sign text (A.2.95)  | O                  |   |
|  |                 | Sign text font (A.2.96)   | O                  | Default enumerators (B.1.7)   |
|  |                 | Sign text – confidence (A.2.97)   | O                  |   |
|  |                 | Number of valid entity states (A.2.98)  | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Entity state (A.2.99)                                      |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid entity states (A.2.98)        |                    |   |



| LSG  | RL LSG<br>M/C/O               | Signal  | RL signal<br>M/C/O | Option  |
|--|-------------------------------|---|--------------------|---|
|  |                               | Entity state (A.2.99)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|  |                               | Sign geometry (A.2.132)   | O                  | Default enumerators (B.1.7)   |
| ☐ ☐ Traffic main sign message colour tone  | C (B.2.1)<br>Relevant: camera | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |   |
|  |                               | Colour value – object level (A.2.100)   | M                  | Profile: Colour model for SOI (7.5.3)   |
|  |                               | Colour tone – confidence – object level (A.2.101)   | O                  | Profile: Colour model for SOI (7.5.3)   |
| ☐ Traffic main sign background colour tone | C (B.2.1)<br>Relevant: camera | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |   |
|  |                               | Colour value – object level (A.2.100)   | M                  | Profile: Colour model for SOI (7.5.3)   |
|  |                               | Colour tone – confidence – object level (A.2.101)   | O                  | Profile: Colour model for SOI (7.5.3)   |
| ☐ Traffic main sign lane relevance         | O                             | Number of valid lane relevance classifications (A.2.133)  | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Lane relevance classification type (A.2.134) |
|  |                               | Size type: dynamic/fixed<br>Size #: Number of valid lane relevance classifications (A.2.133)                              |                    |   |
|  |                               | Lane relevance classification type (A.2.134)  | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|  |                               | Lane relevance classification type – confidence (A.2.135)   | M                  |   |



| LSG                              | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O                          | Option   |
|----------------------------------|-----------------|---|---|--|
| ☐ Traffic main sign position     | M               | Reference point (A.2.14)  | 0   | Default enumerators (B.1.7)  |
|                                  |                 | Position – object level {x, y, z} (A.2.15)  | {M, M, O}                                   |  |
|                                  |                 | Position – object level {x, y, z} – error (A.2.16)  | {M, M, O}                                   | Implementation EM (B.4.1)  |
|                                  |                 | Orientation {yaw, pitch, roll} (A.2.17)   | {C (B.2.1)<br>Relevant:<br>camera, O,<br>O} |  |
|                                  |                 | Orientation {yaw, pitch, roll} – error (A.2.18)   | 0   | Implementation EM (B.4.1)  |
| ☐ Traffic main sign bounding box | O               | Bounding box extent {length, width, height} (A.2.20)                                      | {O, M, M}                                   |  |
|                                  |                 | Bounding box extent {length, width, height} – error (A.2.21)                              | 0   | Implementation EM (B.4.1)  |
| ☐ Supplementary signs            | M               | Number of valid traffic supplementary signs (A.2.136)                                     | M   | Optimise LL (B.1.1)  |
|                                  |                 | Size type: dynamic/fixed<br>Size #: Number of valid traffic supplementary signs (A.2.136) |   |  |
| ☐ ☐ Supplementary sign status    | M               | Existence probability – object level (A.2.1)  | M   |  |
|                                  |                 | Age (A.2.4)   | M   |  |
|                                  |                 | Number of valid observations – object level (A.2.5)                                       | 0   | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|                                  |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5)   |   |  |
|                                  |                 | Time stamp reference – object level (A.2.6)   | M   | Alternative VRO (B.1.3)  |







| LSG  | RL LSG<br>M/C/O | Signal |  | RL signal<br>M/C/O   | Option  |
|--|-----------------|--------|--|--|---|
|  |                 |        |  | M  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)   |
|  |                 |        |  | O  | Track quality (A.2.8)   |
|  |                 |        |  | M  | Measurement status – object level (A.2.9)<br>Default enumerators (B.1.7)  |
| <input type="checkbox"/> <input type="checkbox"/> Supplementary sign information | M               |        |  | M  | Number of valid supplementary sign classifications (A.2.137)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Supplementary sign classification type (A.2.138) |
|  |                 |        |  | Size type: dynamic/fixed<br>Size #: Number of valid supplementary sign classifications (A.2.137) |   |
|  |                 |        |  | M  | Supplementary sign classification type (A.2.138)<br>Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|  |                 |        |  | M  | Supplementary sign classification type – confidence (A.2.139)   |
|  |                 |        |  | M  | Sign value (A.2.93)   |
|  |                 |        |  | M  | Sign value unit (A.2.94)<br>Default enumerators (B.1.7)   |
|  |                 |        |  | O  | Sign text (A.2.95)  |
|  |                 |        |  | O  | Sign text font (A.2.96)<br>Default enumerators (B.1.7)  |
|  |                 |        |  | O  | Sign text – confidence (A.2.97)   |
|  |                 |        |  | M  | Number of valid entity states (A.2.98)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Entity state (A.2.99)  |

| LSG   | RL LSG<br>M/C/O                    | Signal  | RL signal<br>M/C/O | Option  |
|---|------------------------------------|---|--------------------|---|
|   |                                    | Size type: dynamic/fixed<br>Size #: Number of valid entity states (A.2.98)  |                    |   |
|   |                                    | Entity state (A.2.99)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|   <br>Supplementary sign message colour tone | C (B.2.1)<br>Relevant: camera      | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |   |
|   |                                    | Colour value – object level (A.2.100)   | M                  | Profile: Colour model for SOI (7.5.3)   |
|   |                                    | Colour tone – confidence – object level (A.2.101)   | O                  | Profile: Colour model for SOI (7.5.3)   |
|  <br>Supplementary sign background colour tone  | C (B.2.1)<br>Relevant: camera      | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |   |
|   |                                    | Colour value – object level (A.2.100)   | M                  | Profile: Colour model for SOI (7.5.3)   |
|   |                                    | Colour tone – confidence – object level (A.2.101)   | O                  | Profile: Colour model for SOI (7.5.3)   |
|  <br>Supplementary sign position  | M                                  | Relative position (A.2.140)   | M                  | Default enumerators (B.1.7)   |
|   |                                    | Relative position order (A.2.141)   | M                  |   |
| <b>Traffic sign boards</b>  | C (B.2.3)<br>multiple entity types | Recognised traffic sign boards – capability (A.1.10.6)  | O                  |   |
|   |                                    | Recognised traffic sign boards – status (A.1.11.6)  | O                  | Redundancy (B.1.2)<br>Signal: Recognised traffic sign boards – capability (A.1.10.6)<br>Default enumerators (B.1.7) |
|   |                                    | Number of valid traffic sign boards (A.1.12.6)  | M                  | Optimise LL (B.1.1)   |
|   |                                    | Size type: dynamic/fixed<br>Size #: Number of valid traffic sign boards (A.1.12.6)  |                    |   |

| LSG                              | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|----------------------------------|-----------------|---|--------------------|--|
| ☐ Traffic sign board status      | M               | Existence probability – object level (A.2.1)  | M                  |  |
|                                  |                 | Object ID (A.2.2)   | M                  | Alternative A2I (B.3.2)  |
|                                  |                 | Object grouping ID (A.2.3)  | O                  |  |
|                                  |                 | Age (A.2.4)   | M                  |  |
|                                  |                 | Number of valid observations – object level (A.2.5)                                     | O                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|                                  |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5) |                    |  |
|                                  |                 | Time stamp reference – object level (A.2.6)   | M                  | Alternative VRO (B.1.3)  |
|                                  |                 | Observation status – object level (A.2.7)   | M                  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
|                                  |                 | Track quality (A.2.8)   | O                  |  |
|                                  |                 | Measurement status – object level (A.2.9)   | M                  | Default enumerators (B.1.7)  |
| ☐ Traffic sign board information | O               | Number of valid entity states (A.2.98)  | O                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Entity state (A.2.99)   |
|                                  |                 | Size type: dynamic/fixed<br>Size #: Number of valid entity states (A.2.98)              |                    |  |







| LSG                              | RL LSG<br>M/C/O                                       | Signal  | RL signal<br>M/C/O | Option   |
|----------------------------------|---|---|--------------------|--|
|                                  |   | Entity state (A.2.99)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)                               |
| ☐ Traffic sign board colour tone | C (B.2.1)<br>Relevant: camera                         | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |  |
|                                  |   | Colour value – object level (A.2.100)   | M                  | Profile: Colour model for SOI (7.5.3)  |
|                                  |   | Colour tone – confidence – object level (A.2.101)   | O                  | Profile: Colour model for SOI (7.5.3)  |
| ☐ Traffic sign board position    | M   | Position uv_origin {x, y, z} (A.2.142)  | M                  | Profile: Detection references for 3D detections (7.5.4)                              |
|                                  |   | Position uv_origin {x, y, z} – error (A.2.143)  | M                  | Profile: Detection references for 3D detections (7.5.4)<br>Implementation EM (B.4.1) |
|                                  |   | Position u_end {x, y, z} (A.2.144)  | M                  | Profile: Detection references for 3D detections (7.5.4)                              |
|                                  |   | Position u_end {x, y, z} – error (A.2.145)  | M                  | Profile: Detection references for 3D detections (7.5.4)<br>Implementation EM (B.4.1) |
|                                  |   | Position v_end {x, y, z} (A.2.146)  | M                  | Profile: Detection references for 3D detections (7.5.4)                              |
|                                  |   | Position v_end {x, y, z} – error (A.2.147)  | M                  | Profile: Detection references for 3D detections (7.5.4)<br>Implementation EM (B.4.1) |
|                                  |   | Ratio U- to V-axis (A.2.148)  | O                  |  |
|                                  |   | Ratio U- to V-axis – error (A.2.149)  | O                  | Implementation EM (B.4.1)  |
| ☐ Traffic sign board regions     | C (B.2.3)<br>multiple traffic sign board entity types | Number of valid traffic sign board regions (A.2.150)  | M                  | Optimise LL (B.1.1)  |
|                                  |   | Size type: dynamic/fixed<br>Size #: Number of valid traffic sign board regions (A.2.150)                                  |                    |  |






| LSG   | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O | Option  |
|---|-----------------|--|--------------------|---|
|   Traffic sign board<br>region status | M               | Existence probability –<br>object level (A.2.1)  | M                  |   |
|   |                 | Age (A.2.4)  | M                  |   |
|   |                 | Number of valid<br>observations – object level<br>(A.2.5)  | O                  | Optimise LL<br>(B.1.1)<br>Alternative UTL<br>(B.1.6)<br>Key: Time stamp<br>reference – object<br>level (A.2.6)<br>* n recent<br>messages of the<br>interface [see<br>“Time stamp<br>reference – object<br>level” (A.2.6)] |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5)              |                    |   |
|   |                 | Time stamp reference –<br>object level (A.2.6)   | M                  | Alternative VRO<br>(B.1.3)  |
|   |                 | Observation status –<br>object level (A.2.7)   | M                  | Redundancy<br>(B.1.2)<br>Signal: Object ID<br>(A.2.2), Time<br>stamp – prediction<br>(A.1.5.1)<br>Default<br>enumerators<br>(B.1.7)   |
|   |                 | Track quality (A.2.8)  | O                  |   |
|   |                 | Measurement status – object<br>level (A.2.9)   | M                  | Default<br>enumerators<br>(B.1.7)   |
|   |                 | Sign board entity ID<br>(A.2.151)  | O                  | Alternative A2I<br>(B.3.2)  |
|   |                 | Number of valid entity ID<br>references – object level<br>(A.2.152)                                  | O                  | Optimise LL<br>(B.1.1)<br>Alternative A2I<br>(B.3.2)  |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid entity ID references – object level<br>(A.2.152) |                    |   |
|   |                 | Sign board entity ID<br>reference – object level<br>(A.2.153)  | M                  |   |

| LSG   | RL LSG<br>M/C/O               | Signal  | RL signal<br>M/C/O | Option  |
|---|-------------------------------|---|--------------------|---|
|   Traffic sign board region information       | 0                             | Number of valid entity states (A.2.98)  | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Entity state (A.2.99)      |
|   |                               | Size type: dynamic/fixed<br>Size #: Number of valid entity states (A.2.98)  |                    |   |
|   |                               | Entity state (A.2.99)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)                            |
|   |                               | Border width {u, v} (A.2.154)   | 0                  |   |
|   |                               | Border width {u, v} – error (A.2.155)   | 0                  | Implementation EM (B.4.1)   |
|   |                               | Traffic sign board region – confidence (A.2.156)  | 0                  |   |
|   Traffic sign board region colour tone   | C (B.2.1)<br>Relevant: camera | Number of valid colour tones (A.2.157)  | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Colour tone type (A.2.158) |
|   |                               | Size type: dynamic/fixed<br>Size #: Number of valid colour tones (A.2.157)  |                    |   |
|   |                               | Colour tone type (A.2.158)  | M                  | Alternative VRO (B.1.3)   |
|   |                               | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |   |
|   |                               | Colour value – object level (A.2.100)   | M                  | Profile: Colour model for SOI (7.5.3)   |
|   |                               | Colour tone – confidence – object level (A.2.101)   | 0                  | Profile: Colour model for SOI (7.5.3)   |
|   Traffic sign board region vertex points | M                             | Number of valid vertex points (A.2.159)   | M                  | Optimise LL (B.1.1)   |
|   |                               | Size type: dynamic/fixed<br>Size #: Number of valid vertex points (A.2.159)   |                    |   |
|   |                               | Local position {u, v} (A.2.160)   | M                  |   |





| LSG   | RL LSG<br>M/C/O                                    | Signal  | RL signal<br>M/C/O | Option   |
|---|--|---|--------------------|--|
|   |  | Local position {u, v} – error (A.2.161)   | M                  | Implementation EM (B.4.1)  |
| <input type="checkbox"/> Traffic sign board guiding graphs                                | C (B.2.3) multiple traffic sign board entity types | Number of valid traffic sign board guiding graphs (A.2.162)                                     | M                  | Optimise LL (B.1.1)  |
| <input type="checkbox"/> <input type="checkbox"/> Traffic sign board guiding graph status | M  | Size type: dynamic/fixed<br>Size #: Number of valid traffic sign board guiding graphs (A.2.162) |                    |  |
|   |  | Existence probability – object level (A.2.1)  | M                  |  |
|   |  | Age (A.2.4)   | M                  |  |
|   |  | Number of valid observations – object level (A.2.5)   | O                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|   |  | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5)         |                    |  |
|   |  | Time stamp reference – object level (A.2.6)   | M                  | Alternative VRO (B.1.3)  |
|   |  | Observation status – object level (A.2.7)   | M                  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
|   |  | Track quality (A.2.8)   | O                  |  |
|   |  | Measurement status – object level (A.2.9)   | M                  | Default enumerators (B.1.7)  |
|   |  | Sign board entity ID (A.2.151)  | O                  | Alternative A2I (B.3.2)  |
|   |  | Number of valid entity ID references – object level (A.2.152)                                   | O                  | Optimise LL (B.1.1)<br>Alternative A2I (B.3.2)   |

| LSG  | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|--|-----------------|---|--------------------|--|
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid entity ID references – object level (A.2.152) |                    |  |
|  |                 | Sign board entity ID reference – object level (A.2.153)   | M                  |  |
|   Traffic sign board guiding graph information   | O               | Traffic sign board guiding graph – confidence (A.2.163)   | M                  |  |
|   Traffic sign board guiding graph vertex points | M               | Number of valid vertex points (A.2.159)   | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Local position ID (A.2.164) |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid vertex points (A.2.159)                       |                    |  |
|  |                 | Local position ID (A.2.164)   | M                  | Alternative VRO (B.1.3)  |
|  |                 | Local position {u, v} (A.2.160)   | M                  |  |
|  |                 | Local position {u, v} – error (A.2.161)   | M                  | Implementation EM (B.4.1)  |
|   Traffic sign board guiding graph edges     | M               | Number of valid edges (A.2.165)   | M                  | Optimise LL (B.1.1)  |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid edges (A.2.165)                               |                    |  |
|  |                 | Local position ID reference {begin, end} (A.2.166)  | M                  |  |
|  |                 | Edge type (A.2.167)   | M                  | Default enumerators (B.1.7)  |
|  |                 | Edge width {u, v} (A.2.168)   | O                  |  |
|  |                 | Edge width {u, v} – error (A.2.169)   | O                  | Implementation EM (B.4.1)  |
|  |                 | Border width {u, v} (A.2.154)   | O                  |  |
|  |                 | Border width {u, v} – error (A.2.155)   | O                  | Implementation EM (B.4.1)  |
|  |                 | Edge curve type (A.2.170)   | O                  | Default enumerators (B.1.7)  |

| LSG   | RL LSG<br>M/C/O                                    | Signal  | RL signal<br>M/C/O | Option  |
|---|--|---|--------------------|---|
|   |  | Arrow type (A.2.171)  | O                  | Default enumerators (B.1.7)   |
|   |  | Number of valid entity states (A.2.98)  | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Entity state (A.2.99)      |
|   |  | Size type: dynamic/fixed<br>Size #: Number of valid entity states (A.2.98)  |                    |   |
|   |  | Entity state (A.2.99)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)                            |
|   Traffic sign board guiding graph edge colour tone | M  | Number of valid colour tones (A.2.157)  | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Colour tone type (A.2.158) |
|   |  | Size type: dynamic/fixed<br>Size #: Number of valid colour tones (A.2.157)  |                    |   |
|   |  | Colour tone type (A.2.158)  | M                  | Alternative VRO (B.1.3)   |
|   |  | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |   |
|   |  | Colour value – object level (A.2.100)   | M                  | Profile: Colour model for SOI (7.5.3)   |
|   |  | Colour tone – confidence – object level (A.2.101)   | O                  | Profile: Colour model for SOI (7.5.3)   |
|  Traffic sign board text   | C (B.2.3) multiple traffic sign board entity types | Number of valid traffic sign board texts (A.2.172)  | M                  | Optimise LL (B.1.1)   |
|   |  | Size type: dynamic/fixed<br>Size #: Number of valid traffic sign board texts (A.2.172)                                    |                    |   |
|   Traffic sign board text status                | M  | Existence probability – object level (A.2.1)  | M                  |   |
|   |  | Age (A.2.4)   | M                  |   |

| LSG                                     | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|---|-----------------|---|--------------------|--|
|   |                 | Number of valid observations – object level (A.2.5)   | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5)           |                    |  |
|   |                 | Time stamp reference – object level (A.2.6)   | M                  | Alternative VRO (B.1.3)  |
|   |                 | Observation status – object level (A.2.7)   | M                  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
|   |                 | Track quality (A.2.8)   | 0                  |  |
|   |                 | Measurement status – object level (A.2.9)   | M                  | Default enumerators (B.1.7)  |
|   |                 | Sign board entity ID (A.2.151)  | 0                  | Alternative A2I (B.3.2)  |
|   |                 | Number of valid entity ID references – object level (A.2.152)                                     | 0                  | Optimise LL (B.1.1)<br>Alternative A2I (B.3.2)   |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid entity ID references – object level (A.2.152) |                    |  |
|   |                 | Sign board entity ID reference – object level (A.2.153)   | M                  |  |
| ☐ ☐ Traffic sign board text information | M               | Sign text (A.2.95)  | M                  |  |
|   |                 | Sign text font (A.2.96)   | 0                  | Default enumerators (B.1.7)  |

| LSG  | RL LSG<br>M/C/O                                    | Signal  | RL signal<br>M/C/O | Option   |
|--|--|---|--------------------|--|
|  |  | Sign text – confidence (A.2.97)   | 0                  |  |
|  |  | Number of valid entity states (A.2.98)  | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Entity state (A.2.99) |
|  |  | Size type: dynamic/fixed<br>Size #: Number of valid entity states (A.2.98)  |                    |  |
|  |  | Entity state (A.2.99)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)                       |
| ☐ ☐ Traffic sign board text colour tone      | M  | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |  |
|  |  | Colour value – object level (A.2.100)   | M                  | Profile: Colour model for SOI (7.5.3)  |
|  |  | Colour tone – confidence – object level (A.2.101)   | 0                  | Profile: Colour model for SOI (7.5.3)  |
| ☐ ☐ Traffic sign board text bounding box     | M  | Bounding box {u <sub>Begin</sub> , u <sub>End</sub> , v <sub>Begin</sub> , v <sub>End</sub> } (A.2.173)                   | M                  |  |
|  |  | Bounding box {u <sub>Begin</sub> , u <sub>End</sub> , v <sub>Begin</sub> , v <sub>End</sub> } – error (A.2.174)           | 0                  | Implementation EM (B.4.1)  |
| ☐ Traffic sign board icons and symbols       | C (B.2.3) multiple traffic sign board entity types | Number of valid traffic sign board icons and symbols (A.2.175)  | M                  | Optimise LL (B.1.1)  |
|  |  | Size type: dynamic/fixed<br>Size #: Number of valid traffic sign board icons and symbols (A.2.175)                        |                    |  |
| ☐ ☐ Traffic sign board icon or symbol status | M  | Existence probability – object level (A.2.1)  | M                  |  |
|  |  | Age (A.2.4)   | M                  |  |




| LSG   | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|---|-----------------|---|--------------------|--|
|   |                 | Number of valid observations – object level (A.2.5)   | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5)           |                    |  |
|   |                 | Time stamp reference – object level (A.2.6)   | M                  | Alternative VRO (B.1.3)  |
|   |                 | Observation status – object level (A.2.7)   | M                  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
|   |                 | Track quality (A.2.8)   | 0                  |  |
|   |                 | Measurement status – object level (A.2.9)   | M                  | Default enumerators (B.1.7)  |
|   |                 | Sign board entity ID (A.2.151)  | 0                  | Alternative A2I (B.3.2)  |
|   |                 | Number of valid entity ID references – object level (A.2.152)                                     | 0                  | Optimise LL (B.1.1)<br>Alternative A2I (B.3.2)   |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid entity ID references – object level (A.2.152) |                    |  |
|   |                 | Sign board entity ID reference – object level (A.2.153)   | M                  |  |
|   Traffic sign board icon or symbol information | M               | Icon and symbol type (A.2.176)  | M                  | Default enumerators (B.1.7)  |
|   |                 | Icon and symbol type – confidence (A.2.177)   | 0                  |  |

| LSG  | RL LSG<br>M/C/O                                    | Signal   | RL signal<br>M/C/O | Option   |
|--|--|--|--------------------|--|
|  |  | Number of valid entity states (A.2.98)   | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Entity state (A.2.99)   |
|  |  | Size type: dynamic/fixed<br>Size #: Number of valid entity states (A.2.98)                         |                    |  |
|  |  | Entity state (A.2.99)  | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)   |
| ☐ ☐ Traffic sign board icon or symbol bounding box | M  | Bounding box {uBegin, uEnd, vBegin, vEnd} (A.2.173)  | M                  |  |
|  |  | Bounding box {uBegin, uEnd, vBegin, vEnd} – error (A.2.174)  | 0                  | Implementation EM (B.4.1)  |
| ☐ Traffic sign board object references             | C (B.2.3) multiple traffic sign board entity types | Number of valid traffic sign board object references (A.2.178)                                     | M                  | Optimise LL (B.1.1)  |
|  |  | Size type: dynamic/fixed<br>Size #: Number of valid traffic sign board object references (A.2.178) |                    |  |
| ☐ ☐ Traffic sign board object reference status     | M  | Existence probability – object level (A.2.1)   | M                  |  |
|  |  | Age (A.2.4)  | M                  |  |
|  |  | Number of valid observations – object level (A.2.5)  | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|  |  | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5)            |                    |  |
|  |  | Time stamp reference – object level (A.2.6)  | M                  | Alternative VRO (B.1.3)  |

| LSG  | RL LSG<br>M/C/O                                    | Signal  | RL signal<br>M/C/O | Option  |
|--|--|---|--------------------|---|
|  |  | Observation status – object level (A.2.7)   | M                  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7) |
|  |  | Track quality (A.2.8)   | O                  |   |
|  |  | Measurement status – object level (A.2.9)   | M                  | Default enumerators (B.1.7)   |
|  |  | Sign board entity ID (A.2.151)  | O                  | Alternative A2I (B.3.2)   |
|  |  | Number of valid entity ID references – object level (A.2.152)   | O                  | Optimise LL (B.1.1)<br>Alternative A2I (B.3.2)  |
| <input type="checkbox"/> <input type="checkbox"/> Traffic sign board object reference information  | M  | Size type: dynamic/fixed<br>Size #: Number of valid entity ID references – object level (A.2.152)               |                    |   |
|  |  | Sign board entity ID reference – object level (A.2.153)   | M                  |   |
|  |  | Object ID reference – object level (A.2.179)  | M                  | <sup>a</sup>  |
| <input type="checkbox"/> <input type="checkbox"/> Traffic sign board object reference bounding box | M  | Object ID reference – confidence (A.2.180)  | O                  |   |
|  |  | Bounding box {u <sub>Begin</sub> , u <sub>End</sub> , v <sub>Begin</sub> , v <sub>End</sub> } (A.2.173)         | M                  |   |
| <input type="checkbox"/> Traffic sign board occlusion  | C (B.2.3) multiple traffic sign board entity types | Bounding box {u <sub>Begin</sub> , u <sub>End</sub> , v <sub>Begin</sub> , v <sub>End</sub> } – error (A.2.174) | O                  | Implementation EM (B.4.1)   |
|  |  | Number of valid traffic sign board occlusions (A.2.181)   | M                  | Optimise LL (B.1.1)   |
| <input type="checkbox"/> <input type="checkbox"/> Traffic sign board occlusion status              | M  | Size type: dynamic/fixed<br>Size #: Number of valid traffic sign board occlusions (A.2.181)                     |                    |   |
|  |  | Existence probability – object level (A.2.1)  | M                  |   |
|  |  | Age (A.2.4)   | M                  |   |









| LSG  | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|--|-----------------|---|--------------------|--|
|  |                 | Number of valid observations – object level (A.2.5)   | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5)           |                    |  |
|  |                 | Time stamp reference – object level (A.2.6)   | M                  | Alternative VRO (B.1.3)  |
|  |                 | Observation status – object level (A.2.7)   | M                  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
|  |                 | Track quality (A.2.8)   | 0                  |  |
|  |                 | Measurement status – object level (A.2.9)   | M                  | Default enumerators (B.1.7)  |
|  |                 | Sign board entity ID (A.2.151)  | 0                  | Alternative A2I (B.3.2)  |
|  |                 | Number of valid entity ID references – object level (A.2.152)                                     | 0                  | Optimise LL (B.1.1)<br>Alternative A2I (B.3.2)   |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid entity ID references – object level (A.2.152) |                    |  |
|  |                 | Sign board entity ID reference – object level (A.2.153)   | M                  |  |
| ☐ ☐ Traffic sign board occlusion information | 0               | Occlusion type (A.2.182)  | M                  | Default enumerators (B.1.7)  |
|  |                 | Occlusion type – confidence (A.2.183)   | 0                  |  |

| LSG  | RL LSG<br>M/C/O                 | Signal  | RL signal<br>M/C/O | Option   |
|--|---------------------------------|---|--------------------|--|
|   Traffic sign board occlusion vertex points | M                               | Number of valid vertex points (A.2.159)   | M                  | Optimise LL (B.1.1)  |
|  |                                 | Size type: dynamic/fixed<br>Size #: Number of valid vertex points (A.2.159)             |                    |  |
|  |                                 | Local position {u, v} (A.2.160)   | M                  |  |
|  |                                 | Local position {u, v} – error (A.2.161)   | M                  | Implementation EM (B.4.1)  |
| <b>Traffic lights</b>  | C (B.2.3) multiple entity types | Recognised traffic lights – capability (A.1.10.7)                                       | O                  |  |
|  |                                 | Recognised traffic lights – status (A.1.11.7)   | O                  | Redundancy (B.1.2)<br>Signal: Recognised traffic lights – capability (A.1.10.7)<br>Default enumerators (B.1.7)   |
|  |                                 | Number of valid traffic lights (A.1.12.7)   | M                  | Optimise LL (B.1.1)  |
|  |                                 | Size type: dynamic/fixed<br>Size #: Number of valid traffic lights (A.1.12.7)           |                    |  |
|  Traffic light status   | M                               | Existence probability – object level (A.2.1)  | M                  |  |
|  |                                 | Object ID (A.2.2)   | M                  | Alternative A2I (B.3.2)  |
|  |                                 | Object grouping ID (A.2.3)  | O                  |  |
|  |                                 | Age (A.2.4)   | M                  |  |
|  |                                 | Number of valid observations – object level (A.2.5)                                     | O                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|  |                                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5) |                    |  |
|  |                                 | Time stamp reference – object level (A.2.6)   | M                  | Alternative VRO (B.1.3)  |

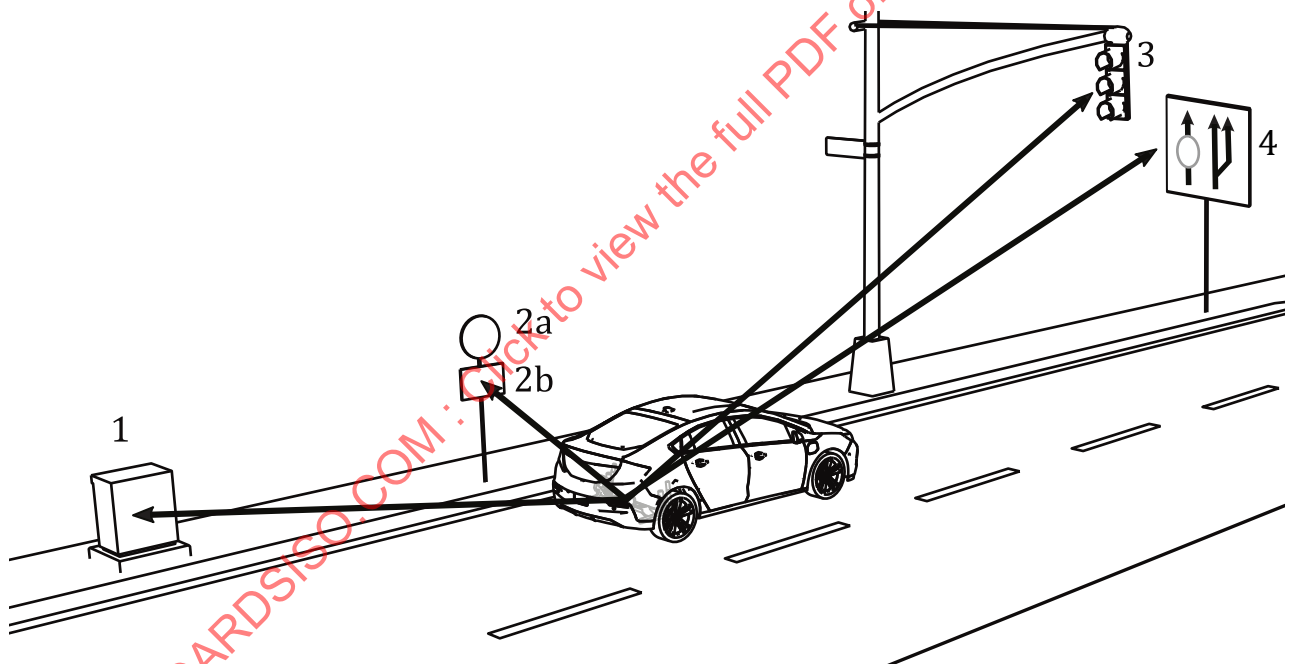
| LSG                         | RL LSG<br>M/C/O        | Signal  | RL signal<br>M/C/O                    | Option  |
|-----------------------------|------------------------|---|---------------------------------------|---|
|                             |                        | Observation status – object level (A.2.7)   | M                                     | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7) |
|                             |                        | Track quality (A.2.8)   | O                                     |   |
|                             |                        | Measurement status – object level (A.2.9)   | M                                     | Default enumerators (B.1.7)   |
| ☐ Traffic light information | M                      | Number of valid structure light classifications (A.2.184)                                     | M                                     | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Structure light classification type (A.2.185)              |
|                             |                        | Size type: dynamic/fixed<br>Size #: Number of valid structure light classifications (A.2.184) |                                       |   |
|                             |                        | Structure light classification type (A.2.185)   | M                                     | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|                             |                        | Structure light classification type – confidence (A.2.186)                                    | M                                     |   |
| ☐ Traffic light position    | M                      | Reference point (A.2.14)  | O                                     | Default enumerators (B.1.7)   |
|                             |                        | Position – object level {x, y, z} (A.2.15)  | {M, M, O}                             |   |
|                             |                        | Position – object level {x, y, z} – error (A.2.16)  | {M, M, O}                             | Implementation EM (B.4.1)   |
|                             |                        | Orientation {yaw, pitch, roll} (A.2.17)   | {C (B.2.1)<br>Relevant: camera, O, O} |   |
|                             |                        | Orientation {yaw, pitch, roll} – error (A.2.18)   | O                                     | Implementation EM (B.4.1)   |
|                             |                        | Minimum visibility distance (A.2.187)   | M                                     |   |
| ☐ Traffic light bounding    | C (B.2.1)<br>Relevant: | Bounding box extent {length, width, height} (A.2.20)  | {O, M, M}                             |   |

| LSG                           | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|-------------------------------|-----------------|---|--------------------|--|
| box                           | camera          | Bounding box extent {length, width, height} – error (A.2.21)                            | 0                  | Implementation EM (B.4.1)  |
| ☐ Traffic light spots         | M               | Total number of traffic light spots (A.2.188)   | 0                  |  |
|                               |                 | Total number of traffic light spots – confidence (A.2.189)                              | 0                  |  |
|                               |                 | Number of valid traffic light spots (A.2.190)   | M                  | Optimise LL (B.1.1)  |
|                               |                 | Size type: dynamic/fixed<br>Size #: Number of valid traffic light spots (A.2.190)       |                    |  |
| ☐ ☐ Traffic light spot status | M               | Existence probability – object level (A.2.1)  | M                  |  |
|                               |                 | Age (A.2.4)   | M                  |  |
|                               |                 | Number of valid observations – object level (A.2.5)                                     | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|                               |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5) |                    |  |
|                               |                 | Time stamp reference – object level (A.2.6)   | M                  | Alternative VRO (B.1.3)  |
|                               |                 | Observation status – object level (A.2.7)   | M                  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
|                               |                 | Track quality (A.2.8)   | 0                  |  |
|                               |                 | Measurement status – object level (A.2.9)   | M                  | Default enumerators (B.1.7)  |

| LSG  | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O   | Option   |
|--|-----------------|---|--|--|
|   Traffic light spot information | M               | Number of valid light shape classifications (A.2.191)                                     | M  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Light shape classification type (A.2.192) |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid light shape classifications (A.2.191) |  |  |
|  |                 | Light shape classification type (A.2.192)   | M  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)   |
|  |                 | Light shape classification type – confidence (A.2.193)                                    | M  |  |
|  |                 | Light shape value (A.2.194)   | C (B.2.2)<br>* The signal “Light shape classification type” (A.2.192) has “LSCT_CountdownSecond”, “LSCT_CountdownPercent” or a similar enumerator or defined during the system design phase. |  |

| LSG   | RL LSG<br>M/C/O               | Signal   | RL signal<br>M/C/O | Option  |
|---|-------------------------------|--|--------------------|---|
|   Traffic light spot colour       | C (B.2.1)<br>Relevant: camera | Number of valid colour classifications (A.2.195)   | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Colour classification type (A.2.196)         |
|   |                               | Size type: dynamic/fixed<br>Size #: Number of valid colour classifications (A.2.195)     |                    |   |
|   |                               | Colour classification type (A.2.196)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|   |                               | Colour classification type – confidence (A.2.197)  | M                  |   |
|   |                               | Number of valid light mode classifications (A.2.198)                                     | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Light mode classification type (A.2.199)     |
|   |                               | Size type: dynamic/fixed<br>Size #: Number of valid light mode classifications (A.2.198) |                    |   |
|   |                               | Light mode classification type (A.2.199)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|   |                               | Light mode classification type – confidence (A.2.200)                                    | M                  |   |
|   Traffic light spot position | M                             | Position – object level {x, y, z} (A.2.15)   | {M, M, O}          |   |
|   |                               | Position – object level {x, y, z} – error (A.2.16)                                       | {M, M, O}          | Implementation EM (B.4.1)   |
|   |                               | Number of valid lane relevance classifications (A.2.133)                                 | O                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Lane relevance classification type (A.2.134) |

| LSG   | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O | Option   |
|---|-----------------|--|--------------------|--|
|   |                 | Size type: dynamic/fixed   |                    |  |
|   |                 | Size #: Number of valid lane relevance classifications (A.2.133) |                    |  |
|   |                 | Lane relevance classification type (A.2.134)                     | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7) |
|   |                 | Lane relevance classification type – confidence (A.2.135)        | M                  |  |
| <sup>a</sup> Reference only SOI traffic sign or traffic light |                 |  |                    |  |



#### Key


- 1 general landmark
- 2a main sign of a traffic sign
- 2b supplementary sign of a traffic sign
- 3 traffic light (with 3 traffic light spots)
- 4 traffic sign board (with guiding graph and a traffic main sign, which is a referenced traffic sign entity)

**Figure 14 — Example for static objects**

#### 7.5.1 Static object header

Table 17 defines the interface header in subclause “Static object header” (7.5.1) and the changes due to the adaptation in comparison to the generic interface header in 7.2.1. The header of the SOI can contain a list of general landmarks, a list of valid traffic signs and/or a list of valid traffic light entities (see 7.5.2).

Table 17 — Specific signal grouping: Static object header

| Signal  | RL signal   |           | Option   |
|---|---|-----------|--|
| Interface version ID {major, minor, patch} (A.1.1)  | Mandatory   |           | Profile: Uniqueness of interface versioning (6.4)  |
| Interface ID (A.1.2)  | Optional  |           | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)   |
| Number of valid serving sensors (A.1.3)   | Mandatory   |           | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4)  |
|  Sensor ID (A.1.4) | Mandatory   |           | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)   |
| Time stamp – prediction (A.1.5.1)   | Mandatory   |           |  |
| Cycle counter (A.1.6.1)   | Optional  |           | Redundancy (B.1.2)<br>Signal: Time stamp – prediction (A.1.5.1)  |
| Interface cycle time (A.1.7)  | Optional  |           |  |
| Interface cycle time – variation (A.1.8)  | Optional  |           |  |
| Data qualifier (A.1.9)  | Mandatory   |           | Default enumerators (B.1.7)  |
| Colour model type (A.1.14)  | Conditional (B.2.1)<br>Relevant camera              |           | Profile: Colour model for SOI (7.5.3)<br>Alternative VRO (B.1.3)<br>Use enumeration to define colours by defining colour values and the applied colour model for each enumerator.<br>Default enumerators (B.1.7) |
| Recognised general landmarks – capability (A.1.10.4)  | RL LSG conditional (B.2.3)<br>multiple entity types | Optional  |  |
| Recognised general landmarks – status (A.1.11.4)  |   | Optional  | Redundancy (B.1.2)<br>Signal: Recognised general landmarks – capability (A.1.10.4)<br>Default enumerators (B.1.7)  |
| Number of valid general landmarks (A.1.12.4)  |   | Mandatory | Optimise LL (B.1.1)  |
| Recognised traffic signs – capability (A.1.10.5)  |   | Optional  |  |
| Recognised traffic signs – status (A.1.11.5)  |   | Optional  | Redundancy (B.1.2)<br>Signal: Recognised traffic signs – capability (A.1.10.5)<br>Default enumerators (B.1.7)  |
| Number of valid traffic signs (A.1.12.5)  |   | Mandatory | Optimise LL (B.1.1)  |



| Signal   | RL signal | Option  |
|--|-----------|---|
| Recognised traffic sign boards – capability (A.1.10.6) | Optional  |   |
| Recognised traffic sign boards – status (A.1.11.6)     | Optional  | Redundancy (B.1.2)<br>Signal: Recognised traffic sign boards – capability (A.1.10.6)<br>Default enumerators (B.1.7) |
| Number of valid traffic sign boards (A.1.12.6)         | Mandatory | Optimise LL (B.1.1)   |
| Recognised traffic lights – capability (A.1.10.7)      | Optional  |   |
| Recognised traffic lights – status (A.1.11.7)          | Optional  | Redundancy (B.1.2)<br>Signal: Recognised traffic lights – capability (A.1.10.7)<br>Default enumerators (B.1.7)      |
| Number of valid traffic lights (A.1.12.7)              | Mandatory | Optimise LL (B.1.1)   |

The following LSGs, which normally shall be provided by the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor cluster (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

### 7.5.2 Static object entity

Each static object describes a recognised real-world general landmark, traffic sign, traffic sign board or traffic light. They consist of several LSGs. The traffic sign consists of a main sign and none, one or more supplementary signs. A main sign is a traffic sign that can be followed on its own (for example, a stop sign). A supplementary sign is a sign with conditions under which the main sign is valid (for example, under wet conditions only). A traffic sign board may consist of regions on the board, guiding graphs, text strings, icons and symbols, referenced object entities (for example, traffic signs, traffic lights) and occlusions.

A general landmark is described by the following LSGs.

- **Status:** the status describes general information of the tracked general landmark. This information is based on history of the object tracking. Table 18 defines the signal grouping “Static object entity status”. It defines this status LSG of the general landmark and redefines the signal grouping “Generic object level entity status” (see Table 8).
- **Information:** the information describes the general landmark type and confidence.
- **Position:** the position describes the position information of the tracked static object.
- **Bounding box:** the bounding box describes the 3D rectangular hull that encloses the recognised static object.

A traffic sign is described by the following LSGs.

- **Main sign status:** the main sign status describes general information of the tracked main sign. This information is based on history of the object tracking. Table 18 defines the signal grouping “Static object entity status”. It defines this status LSG of the main sign and redefines the signal grouping “Generic object level entity status” (see Table 8).
- **Main sign information:** the main sign information describes the semantic of the main sign.
- **Main sign colour tone:** the main sign colour tone describes visual properties of the main sign’s message text and the background.
- **Main sign lane relevance:** the main sign lane relevance describes the lanes, which are relevant for the main sign and its supplementary signs.
- **Main sign position:** the main sign position describes the geometric position and orientation of the main sign.
- **Main sign bounding box:** the main sign bounding box describes the 3D rectangular hull that encloses the recognised main sign.
- A list of supplementary signs assigned to the main sign:
  - **Supplementary sign status:** the supplementary sign status describes general information of the tracked supplementary sign. This information is based on history of the object tracking and is associated to the main sign. A supplementary sign uses a subset of signals of the signal grouping “Static object entity status” (Table 18) for this LSG.
  - **Supplementary sign information:** the supplementary sign information describes the supplementary sign type and confidence.
  - **Supplementary sign colour tone:** the supplementary sign colour tone describes visual properties of the supplementary sign’s message text and the background.
  - **Supplementary sign position:** the supplementary sign position describes the qualitative position of the supplementary sign with respect to the main sign and other supplementary signs.

A traffic sign board is described by the following LSGs (see key 1 in Figure 15).

- **Traffic sign board status:** the sign board status describes general information of the tracked traffic sign board. This information is based on history of the object tracking. Table 18 defines the signal grouping “Static object entity status”. It defines this status LSG of the sign board and redefines the signal grouping “Generic object level entity status” (see Table 8).
- **Traffic sign board information:** the sign board information describes the general information of the traffic sign board.
- **Traffic sign board colour tone:** the sign board colour tone describes visual properties of the traffic sign board.
- **Traffic sign board position:** the position describes the rectangular shape of the tracked traffic sign board. It defines the affine coordinate system of the traffic sign board.

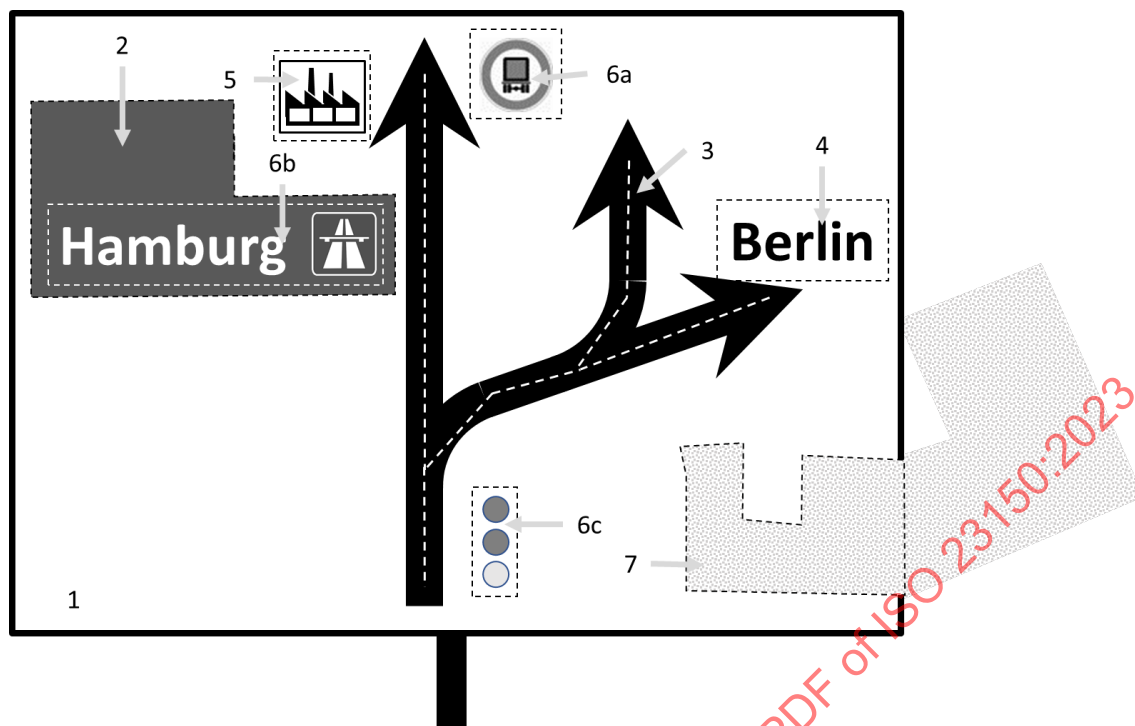
A traffic sign board entity uses multiple traffic sign board entity types as regions, guiding graphs, text, icons and symbols, referenced object entities and occlusions.

- A list of traffic sign board regions which are assigned to the traffic sign board (see key 2 in Figure 15 and Figure 16):
  - **Region status:** the region status describes general information of the tracked region. The region is associated to a sign board type entity or the sign board. A region uses a subset of signals of the signal grouping “Static object entity status” (Table 18) for this LSG and extends signals for a hierarchical association of traffic sign board type entities.
  - **Region information:** the region information describes the information of the region on the traffic sign board.
  - **Region colour tone:** the region colour tone describes visual properties of the region on the traffic sign board.
  - **Region vertex points:** the region vertex points describe the region with a list of relative 2D vertex points for the region by affine coordinates of the traffic sign board object coordinate system.
- A list of traffic sign board guiding graphs which are assigned to the traffic sign board (see key 3 in Figure 15 and Figure 17):
  - **Guiding graph status:** the guiding graph status describes general information of the tracked guiding graph. The guiding graph is associated to a sign board type entity or the sign board. A symbol uses a subset of signals of the signal grouping “Static object entity status” (Table 18) for this LSG and extends signals for a hierarchical association of traffic sign board type entities.
  - **Guiding graph information:** the guiding graph information describes the information of the guiding graph on the traffic sign board.
  - **Guiding graph vertex points:** the guiding graph vertex points describe the guiding graph with a list of relative 2D vertex points by affine coordinates of the traffic sign board object coordinate system.
  - **Guiding graph edges:** the guiding graph edges describe the edges of the guiding graph on the traffic sign board and references vertex points for each edge.
  - **Guiding graph edge colour tone:** the guiding graph edge colour tone describes visual properties of a guiding graph edge on the traffic sign board.
- A list of traffic sign board text strings which are assigned to the traffic sign board (see key 4 in Figure 15), which is no referenced traffic sign (some text strings on the traffic sign board may be recognised as a referenced traffic sign main or supplementary sign object [see LSG “Traffic sign board object references”]):
  - **Text status:** the text status describes general information of the tracked text. The text string is associated to a sign board type entity or the sign board. A text string uses a subset of signals of the signal grouping “Static object entity status” (Table 18) for this LSG and extends signals for a hierarchical association of traffic sign board type entities.
  - **Text information:** the text information describes the text string on the traffic sign board.
  - **Text colour tone:** the text colour tone describes visual properties of the text string on the traffic sign board.

- **Text bounding box:** the text bounding box describes the bounding box of the text string by affine coordinates of the traffic sign board object coordinate system.
- A list of traffic sign board icons and symbols which are assigned to the traffic sign board (see key 5 in Figure 15):
  - **Icon or symbol status:** the icon or symbol status describes general information of the tracked icon or symbol. The icon or symbol is associated to a sign board type entity or the sign board. An icon or symbol uses a subset of signals of the signal grouping “Static object entity status” (Table 18) for this LSG and extends signals for a hierarchical association of traffic sign board type entities.
  - **Icon or symbol information:** the icon or symbol information describes the recognised icon or symbol on the traffic sign board.
  - **Icon or symbol bounding box:** the icon or symbol bounding box describes the bounding box of the icon or the symbol by affine coordinates of the traffic sign board object coordinate system.
- A list of traffic sign board object references for object entities which are assigned to the traffic sign board (for example, traffic sign or traffic light). Due to the possibility of insufficient separability of traffic signs or traffic lights and traffic sign boards that are too close to the traffic sign board, the traffic sign board related object entities are always referenced (see key 6a, 6b and 6c in Figure 15). The possible lack of accuracy in the separation of traffic sign board and traffic signs or traffic lights may link traffic signs or traffic lights to the traffic sign board as object references (see Figure 18):
  - **Object reference status:** the object reference status describes general information of the tracked referenced object entity. The object entity is associated to a sign board type entity or the sign board. An object reference uses a subset of signals of the signal grouping “Static object entity status” (Table 18) for this LSG and extends signals for a hierarchical association of traffic sign board type entities.
  - **Object reference information:** the object reference information describes the semantic of the referenced object entity on the traffic sign board.
  - **Object reference bounding box:** the object reference bounding box describes the bounding box of the referenced object entity by affine coordinates of the traffic sign board object coordinate system.
- A list of traffic sign board occlusion areas which are assigned to the traffic sign board (for example, occlusion by graffiti, a heavy truck, a tree branch) (see key 7 in Figure 15):
  - **Occlusion status:** the occlusion status describes general information of the tracked occlusion area on the traffic sign board. The occlusion is associated to a sign board type entity or the sign board. An occlusion uses a subset of signals of the signal grouping “Static object entity status” (Table 18) for this LSG and extends signals for a hierarchical association of traffic sign board type entities.
  - **Occlusion information:** the occlusion information describes the semantic of the occlusion area on the traffic sign board.
  - **Occlusion vertex points:** the occlusion vertex points describe the occlusion with a list of relative 2D vertex points for the shape of the occlusion by affine coordinates of the traffic sign board object coordinate system.

A traffic light is described by the following LSGs.

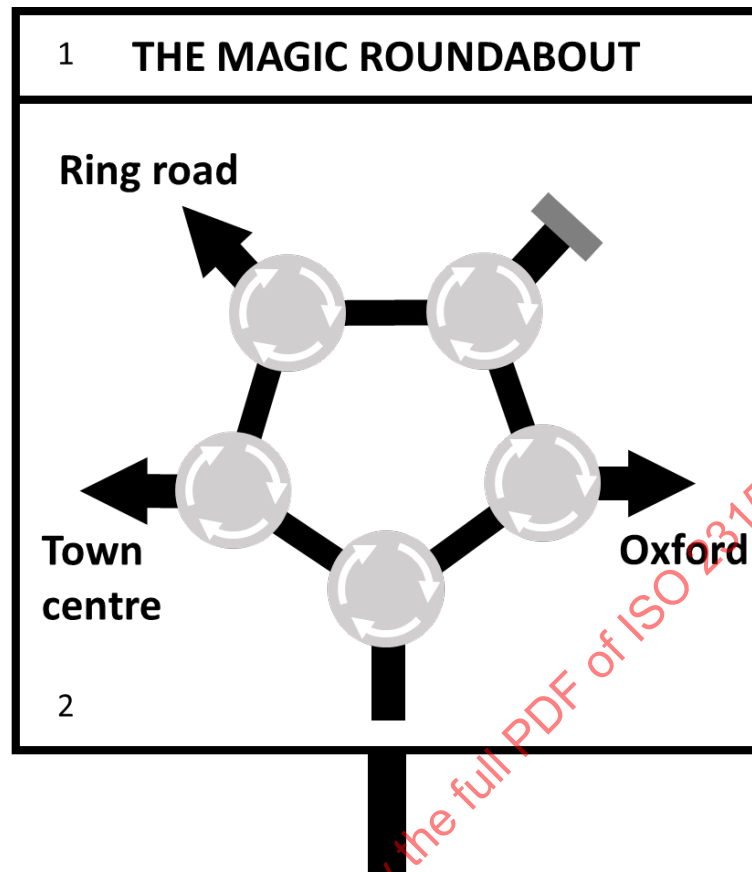
- **Light status:** the traffic light status describes general information of the tracked traffic light. This information is based on history of the object tracking. Table 18 defines the signal grouping “Static object entity status”. It defines this status LSG of the traffic light and redefines the signal grouping “Generic object level entity status” (see Table 8).
- **Light information:** the information describes the certainty and the type of the tracked traffic light.
- **Light position:** the position describes the position information of the tracked traffic light.
- **Light bounding box:** the bounding box describes the 3D rectangular hull that encloses the tracked traffic light.
- A list of traffic light spots assigned to the traffic light:
  - **Spot status:** the traffic light spot status describes general information of the tracked light spot. This information is based on history of the object tracking and is associated to the traffic light. A traffic light spot uses a subset of signals of the signal grouping “Static object entity status” (Table 18) for this LSG.
  - **Spot information:** the traffic light spot information describes the semantic of the tracked light spot.
  - **Spot colour:** the traffic light spot colour describes the basic light colour of the tracked light spot.
  - **Spot position:** the traffic light spot position describes the position of the tracked light spot which is relative to the vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)]. All traffic light spots are inside the traffic light’s bounding box [see signal “Bounding box extent {length, width, height}” (A.2.20)].



#### Key

- 1 traffic sign board  
described by an affine coordinate system
- 2 region on the traffic sign board – region for motorway information  
described by a shape with vertex points
- 3 guiding graph on the traffic sign board  
described by vertex points and connecting edges
- 4 text string on the traffic sign board – text string “Berlin”  
described by a bounding box
- 5 icon or symbol on the traffic sign board – icon for industrial park  
described by a bounding box
- 6a referenced object entity on the traffic sign board – traffic sign as traffic main sign  
described by a bounding box
- 6b referenced object entity on the traffic sign board – traffic sign as traffic main sign including text string “Hamburg”  
described by a bounding box
- 6c referenced object entity on the traffic sign board – traffic light  
described by a bounding box
- 7 occlusion area of the traffic sign board – for example, graffiti or vegetation  
described by a shape with vertex points, which is relevant for the traffic sign board

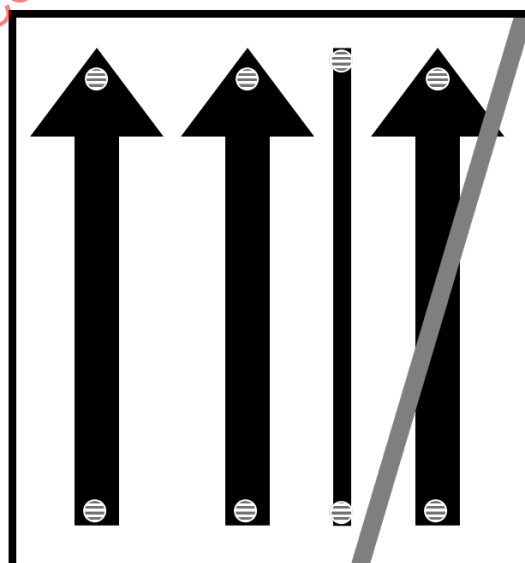
**Figure 15 — Example for a traffic sign board and its specific entities including their bounding box, graph or shape**



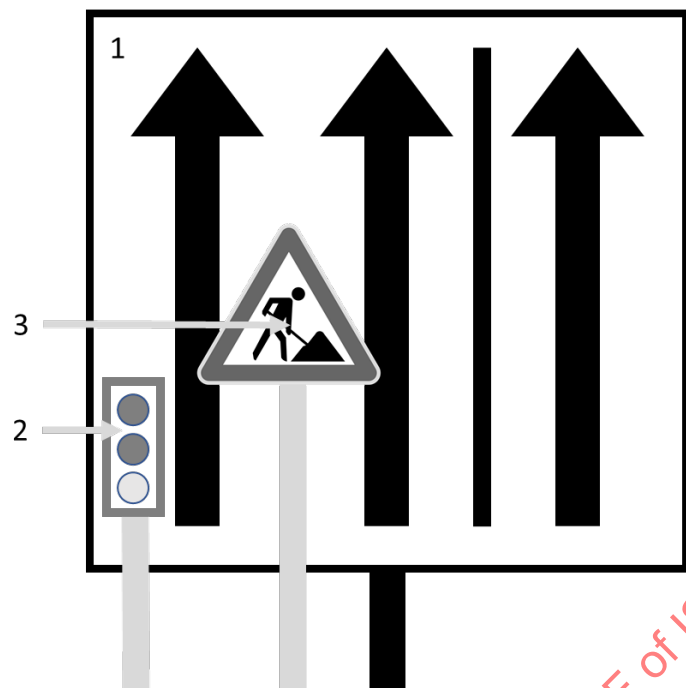
**Key**

- 1 region 1 – title
- 2 region 2 – guiding graph, text and traffic signs, which are referenced traffic sign entities

**Figure 16 — Example of a traffic sign board with regions with borderlines**



**Figure 17 — Example for a traffic sign board with guiding graph and “Entity state” (A.2.99) := enumerator “ES\_PartlyOutOfService”**



Key

- 1 traffic sign board
- 2 traffic light which may be a referenced traffic sign entity of the traffic sign board (see key 1)
- 3 traffic sign which may be a referenced traffic sign entity of the traffic sign board (see key 1)

Figure 18 — Example of a traffic sign board, traffic sign and traffic light which may not be separated exactly

Table 18 — Specific signal grouping: Static object entity status

| Signal  | RL signal | Option   |
|---|-----------|--|
| Existence probability – object level (A.2.1)                                    | Mandatory |  |
| Object ID (A.2.2)   | Mandatory | Alternative A2I (B.3.2)  |
| Object grouping ID (A.2.3)  | Optional  |  |
| Age (A.2.4)   | Mandatory |  |
| Number of valid observations – object level (A.2.5)                             | Optional  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
| <input checked="" type="checkbox"/> Time stamp reference – object level (A.2.6) | Mandatory | Alternative VRO (B.1.3)  |
| <input checked="" type="checkbox"/> Observation status – object level (A.2.7)   | Mandatory | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
| Track quality (A.2.8)   | Optional  |  |
| Measurement status – object level (A.2.9)                                       | Mandatory | Default enumerators (B.1.7)  |



### 7.5.3 Profile: Colour model for SOI

The profile is relevant for camera sensor technology (B.2.1). The signal “Colour model type” (A.1.14) defines the colour model for the interface. Depending on the colour model, each colour is described by a fixed number of colour values [see signal “Colour value – object level” (A.2.100)] to define the colour tone. The confidence of the colour tone is provided by the signal “Colour tone – confidence – object level” (A.2.101).

The interface provides colour tones for traffic main signs, supplementary signs and traffic sign board entity types.


### 7.5.4 Profile: Detection references for 3D detections

As the different sensor types have different strength, some of them are very precise in a coordinate system closer to its nature. To enable the fusion unit to make use of it, a mechanism is provided to map detection of the detection level to entities of the object level.

The profile is relevant, if, for example, a fusion unit benefits from additional information of detections (for example, detections defined in sensor coordinates), which the sensor cluster has used to generate signals on object level.

The confidence of the position (in cartesian coordinates) of a traffic sign board [see signal “Position uv\_origin {x, y, z}” (A.2.142)] and the confidence of the extension of the traffic sign board [see signals “Position u\_end {x, y, z}” (A.2.144) and “Position v\_end {x, y, z}” (A.2.146)] may be low and a fusion unit may combine detections from different sensor technologies. Logical interfaces may extent the signals according to the cross-interface optimisation (see B.3.4) to reference detections to add the additional information of detections (see Table 19). Additionally, the fusion unit may combine detections from different sensor technologies to reduce the corresponding error signals [“Position uv\_origin {x, y, z} – error” (A.2.143), “Position u\_end {x, y, z} – error” (A.2.145) and “Position v\_end {x, y, z} – error” (A.2.147)].

**Table 19 — Signal grouping: Exemplary detection references for traffic sign board affine coordinates**

|   |   |  |   |  |
|---|---|--|---|--|
|  Traffic sign board position | M | Position uv_origin {x, y, z} (A.2.142)         | M | Profile: Detection references for 3D detections (7.5.4)        |
|   |   | Position uv_origin {x, y, z} – error (A.2.143) | M | Profile: Detection references for 3D detections (7.5.4)<br>... |
|   |   | Position u_end {x, y, z} (A.2.144)             | M | Profile: Detection references for 3D detections (7.5.4)        |
|   |   | Position u_end {x, y, z} – error (A.2.145)     | M | Profile: Detection references for 3D detections (7.5.4)<br>... |
|   |   | Position v_end {x, y, z} (A.2.146)             | M | Profile: Detection references for 3D detections (7.5.4)        |

|  |   |  |   |  |
|--|---|--|---|--|
|  |   | Position v_end {x, y, z} – error (A.2.147)   | M | Profile: Detection references for 3D detections (7.5.4)<br>...   |
|  |   | ...  |   |  |
| ☐ Detections to generate signals<br>“Position uv_origin {x, y, z}” (A.2.142) and<br>“Position uv_origin {x, y, z} – error” (A.2.143) | 0 | Number of valid sensing elements which generate the signal “xxx” – Annex B.3.4 (Table B.2) for signals “Position uv_origin {x, y, z}” (A.2.142) and “Position uv_origin {x, y, z} – error” (A.2.143) | M | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Number of valid serving sensors – Annex B.3.4 (Table B.3),<br>Sensor ID reference – Annex B.3.4 (Table B.4) |
|  |   | Size type: dynamic/fixed<br>Size #: Number of valid sensing elements which generate the signal “xxx” – Annex B.3.4 (Table B.2) for signal “Position uv_origin {x, y, z}” (A.2.142)                   |   |  |
|  |   | Number of valid serving sensors – Annex B.3.4 (Table B.3)  | 0 | Alternative VRO (B.1.3)<br>a   |
|  |   | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors – Annex B.3.4 (Table B.3)  |   |  |
|  |   | Sensor ID reference – Annex B.3.4 (Table B.4)  | M | a  |
|  |   | Time stamp – measurement (A.1.5.2)   | 0 | b  |
|  |   | Cycle counter (A.1.6.1)  | 0 | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)<br>b  |
|  |   | Detection ID reference – Annex B.3.4 (Table B.5)   | M |  |
| ☐ Detections to generate signals<br>“Position u_end {x, y, z}” (A.2.144) and<br>“Position u_end {x, y, z} – error” (A.2.145)         | 0 | Number of valid sensing elements which generate the signal “xxx” – Annex B.3.4 (Table B.2) for signals “Position u_end {x, y, z}” (A.2.144) and “Position u_end {x, y, z} – error” (A.2.145)         | M | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Number of valid serving sensors – Annex B.3.4 (Table B.3),<br>Sensor ID reference – Annex B.3.4 (Table B.4) |

|   |   |  |   |   |   |   |   |              |                                    |   |              |                         |   |  |  |   |  |
|---|---|--|---|---|---|---|---|--------------|------------------------------------|---|--------------|-------------------------|---|--|--|---|--|
|   |   | <p>Size type: dynamic/fixed</p> <p>Size #: Number of valid sensing elements which generate the signal "xxx" – Annex B.3.4 (Table B.2) for signal "Position u_end {x, y, z}" (A.2.144)</p> <table> <tr> <td>Number of valid serving sensors – Annex B.3.4 (Table B.3)</td><td>0</td><td>Alternative VRO (B.1.3)<br/><sup>a</sup></td></tr> </table> <p>Size type: dynamic/fixed</p> <p>Size #: Number of valid serving sensors – Annex B.3.4 (Table B.3)</p> <table> <tr> <td>Sensor ID reference – Annex B.3.4 (Table B.4)</td><td>M</td><td><sup>a</sup></td></tr> <tr> <td>Time stamp – measurement (A.1.5.2)</td><td>0</td><td><sup>b</sup></td></tr> <tr> <td>Cycle counter (A.1.6.1)</td><td>0</td><td>Redundancy (B.1.2)<br/>Signal: Time stamp – measurement (A.1.5.2)<br/><sup>b</sup></td></tr> <tr> <td>Detection ID reference – Annex B.3.4 (Table B.5)</td><td>M</td><td></td></tr> </table>   | Number of valid serving sensors – Annex B.3.4 (Table B.3) | 0 | Alternative VRO (B.1.3)<br><sup>a</sup> | Sensor ID reference – Annex B.3.4 (Table B.4) | M | <sup>a</sup> | Time stamp – measurement (A.1.5.2) | 0 | <sup>b</sup> | Cycle counter (A.1.6.1) | 0 | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)<br><sup>b</sup> | Detection ID reference – Annex B.3.4 (Table B.5) | M |  |
| Number of valid serving sensors – Annex B.3.4 (Table B.3)   | 0 | Alternative VRO (B.1.3)<br><sup>a</sup>  |   |   |   |   |   |              |                                    |   |              |                         |   |  |  |   |  |
| Sensor ID reference – Annex B.3.4 (Table B.4)   | M | <sup>a</sup>   |   |   |   |   |   |              |                                    |   |              |                         |   |  |  |   |  |
| Time stamp – measurement (A.1.5.2)  | 0 | <sup>b</sup>   |   |   |   |   |   |              |                                    |   |              |                         |   |  |  |   |  |
| Cycle counter (A.1.6.1)   | 0 | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)<br><sup>b</sup>   |   |   |   |   |   |              |                                    |   |              |                         |   |  |  |   |  |
| Detection ID reference – Annex B.3.4 (Table B.5)  | M |  |   |   |   |   |   |              |                                    |   |              |                         |   |  |  |   |  |
| <p><input checked="" type="checkbox"/> Detections to generate signals "Position v_end {x, y, z}" (A.2.142) and "Position v_end {x, y, z} – error" (A.2.147)</p> | 0 | <p>Number of valid sensing elements which generate the signal "xxx" – Annex B.3.4 (Table B.2) for signals "Position v_end {x, y, z}" (A.2.142) and "Position v_end {x, y, z} – error" (A.2.147)</p> <p>M</p> <p>Optimise LL (B.1.1)<br/>Alternative UTL (B.1.6)<br/>Key: Number of valid serving sensors – Annex B.3.4 (Table B.3),<br/>Sensor ID reference – Annex B.3.4 (Table B.4)</p> <p>Size type: dynamic/fixed</p> <p>Size #: Number of valid sensing elements which generate the signal "xxx" – Annex B.3.4 (Table B.2) for signal "Position v_end {x, y, z}" (A.2.142)</p> <table> <tr> <td>Number of valid serving sensors – Annex B.3.4 (Table B.3)</td><td>0</td><td>Alternative VRO (B.1.3)<br/><sup>a</sup></td></tr> </table> <p>Size type: dynamic/fixed</p> <p>Size #: Number of valid serving sensors – Annex B.3.4 (Table B.3)</p> <table> <tr> <td>Sensor ID reference – Annex B.3.4 (Table B.4)</td><td>M</td><td><sup>a</sup></td></tr> </table> | Number of valid serving sensors – Annex B.3.4 (Table B.3) | 0 | Alternative VRO (B.1.3)<br><sup>a</sup> | Sensor ID reference – Annex B.3.4 (Table B.4) | M | <sup>a</sup> |                                    |   |              |                         |   |  |  |   |  |
| Number of valid serving sensors – Annex B.3.4 (Table B.3)   | 0 | Alternative VRO (B.1.3)<br><sup>a</sup>  |   |   |   |   |   |              |                                    |   |              |                         |   |  |  |   |  |
| Sensor ID reference – Annex B.3.4 (Table B.4)   | M | <sup>a</sup>   |   |   |   |   |   |              |                                    |   |              |                         |   |  |  |   |  |

|   |  |  |  |   |   |
|---|--|--|--|---|---|
|   |  |  | Time stamp – measurement (A.1.5.2)               | 0 | b   |
|   |  |  | Cycle counter (A.1.6.1)                          | 0 | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)<br>b |
|   |  |  | Detection ID reference – Annex B.3.4 (Table B.5) | M |   |
| a Signals are required, if the signal “Detection ID” (A.4.2) is not unique for different DLI of the sensor cluster.     |  |  |  |   |   |
| b Signals are required, if the signal “Detection ID” (A.4.2) is not unique for different measurement cycles of the DLI. |  |  |  |   |   |


The mechanism of the profile based on a traffic sign board definition, but can also be used for traffic light, landmarks, lane markings, static or potentially moving objects.

## 7.6 Free space area object interface

The free space area object interface (FSAOI) consists of one type of areas: free space area (see Figure 19). Table 20 defines the logical structure of the free space area object interface.






**Table 20 — Free space area object interface**

| LSG                       | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O | Option  |
|---------------------------|-----------------|--|--------------------|---|
| Information:<br>interface | M               | Interface version ID {major, minor, patch} (A.1.1) | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)  |
|                           |                 | Interface ID (A.1.2)                               | O                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Default<br>enumerators<br>(B.1.7)   |
|                           |                 | Number of valid serving sensors (A.1.3)            | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Optimise LL<br>(B.1.1)<br>Alternative UTL<br>(B.1.6)<br>Key: Sensor ID<br>(A.1.4) |

| LSG   | RL LSG<br>M/C/O                       | Signal  | RL signal<br>M/C/O | Option   |
|---|---------------------------------------|---|--------------------|--|
|   |                                       | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors (A.1.3)     |                    |  |
|   |                                       | Sensor ID (A.1.4)   | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Alternative VRO<br>(B.1.3)   |
|   |                                       | Time stamp – prediction (A.1.5.1)   | M                  |  |
|   |                                       | Cycle counter (A.1.6.1)   | O                  | Redundancy<br>(B.1.2)<br>Signal: Time stamp<br>– prediction<br>(A.1.5.1)   |
|   |                                       | Interface cycle time (A.1.7)  | O                  |  |
|   |                                       | Interface cycle time – variation<br>(A.1.8)                                     | O                  |  |
|   |                                       | Data qualifier (A.1.9)  | M                  | Default<br>enumerators<br>(B.1.7)  |
| Free space areas  | C (B.2.3)<br>multiple<br>entity types | Recognised free space areas –<br>capability (A.1.10.8)                          | O                  |  |
|   |                                       | Recognised free space areas – status<br>(A.1.11.8)                              | O                  | Redundancy<br>(B.1.2)<br>Signal: Recognised<br>free space areas –<br>capability<br>(A.1.10.8)<br>Default<br>enumerators<br>(B.1.7) |
|   |                                       | Number of valid free space areas<br>(A.1.12.8)                                  | M                  | Optimise LL<br>(B.1.1)   |
|   |                                       | Size type: dynamic/fixed<br>Size #: Number of valid free space areas (A.1.12.8) |                    |  |
|  Free space area<br>status | M                                     | Existence probability – object<br>level (A.2.1)                                 | M                  |  |
|   |                                       | Object ID (A.2.2)   | M                  | Alternative A2I<br>(B.3.2)   |
|   |                                       | Object grouping ID (A.2.3)  | O                  |  |
|   |                                       | Age (A.2.4)   | M                  |  |

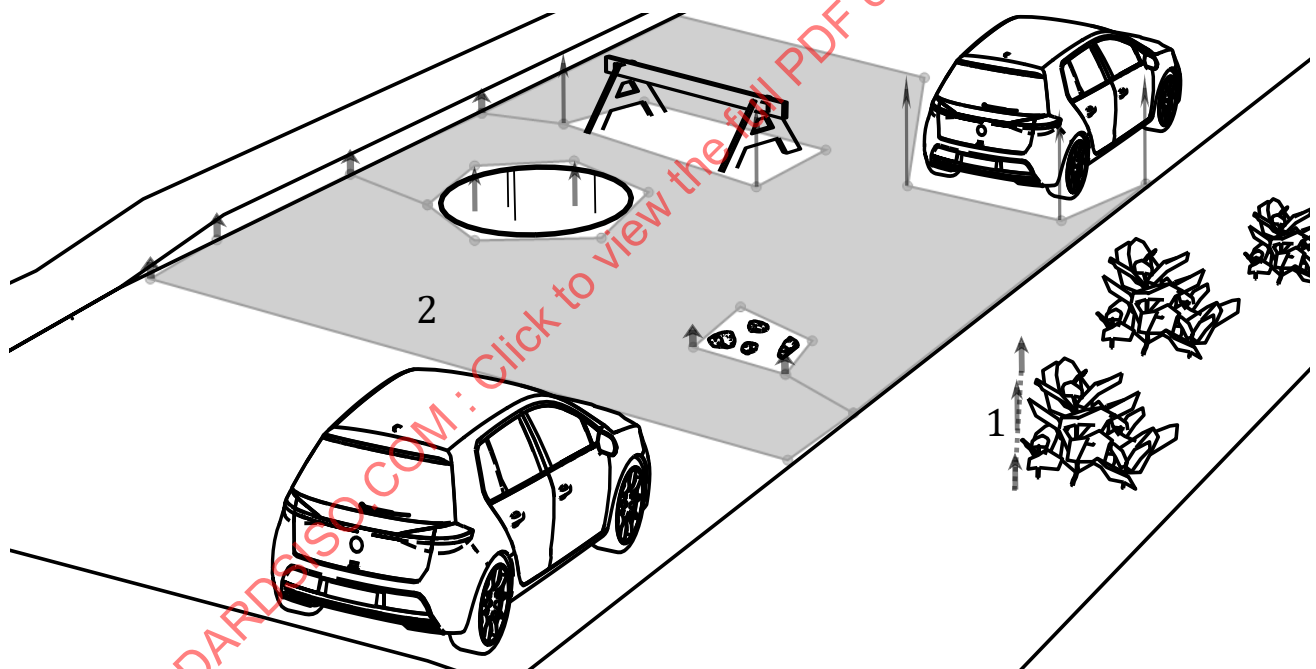
| LSG  | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|--|-----------------|---|--------------------|--|
|  |                 | Number of valid observations – object level (A.2.5)                                     | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – object level (A.2.5) |                    |  |
|  |                 | Time stamp reference – object level (A.2.6)   | M                  | Alternative VRO (B.1.3)  |
|  |                 | Observation status – object level (A.2.7)   | M                  | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
|  |                 | Track quality (A.2.8)   | 0                  |  |
|  |                 | Measurement status – object level (A.2.9)   | M                  | Default enumerators (B.1.7)  |
| <input type="checkbox"/> Free space area information                           | M               | Free space type (A.2.201)   | M                  |  |
|  |                 | Polyline interpolation method (A.2.120)   | M                  | Default enumerators (B.1.7)  |
|  |                 | Number of valid free space area shapes (A.2.202)  | M                  | Optimise LL (B.1.1)  |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid free space area shapes (A.2.202)    |                    |  |
| <input type="checkbox"/> <input type="checkbox"/> Free space area road surface | 0               | Road type – object level (A.2.73)   | M                  | Default enumerators (B.1.7)  |
|  |                 | Road type – confidence (A.2.74)   | M                  |  |

| LSG   | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option  |
|---|-----------------|---|--------------------|---|
|   |                 | Number of valid road surface classifications (A.2.75)   | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Road surface classification type – object level (A.2.76)           |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid road surface classifications (A.2.75)           |                    |   |
|   |                 | Road surface classification type – object level (A.2.76)  | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|   |                 | Road surface classification type – confidence (A.2.77)  | M                  |   |
|   |                 | Road surface roughness – object level (A.2.78)  | 0                  |   |
|   |                 | Road surface texture – object level (A.2.79)  | 0                  |   |
|   |                 | Number of valid road surface condition classifications (A.2.80)                                     | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Road surface condition classification type – object level (A.2.81) |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid road surface condition classifications (A.2.80) |                    |   |
|   |                 | Road surface condition classification type – object level (A.2.81)                                  | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|   |                 | Road surface condition classification type – confidence (A.2.82)                                    | M                  |   |
| <input type="checkbox"/> <input type="checkbox"/> Free space area shape | M               | Limitation geometry type (A.2.203)  | M                  |   |
|   |                 | Shape type – object level (A.2.204)   | M                  | Default enumerators (B.1.7)   |

| LSG   | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O  | Option                            |
|---|-----------------|--|---|-----------------------------------|
|   |                 | Sector common vertex point<br>{x, y, z} (A.2.205)                      | {C (B.2.2), C (B.2.2), 0}<br>* The signal "Shape type – object level" (A.2.204) has "ST_Sector" or a similar enumerator defined during the system design phase. |                                   |
|   |                 | Sector common vertex point<br>{x, y, z} – error (A.2.206)              | {C (B.2.2), C (B.2.2), 0}<br>* The signal "Shape type – object level" (A.2.204) has "ST_Sector" or a similar enumerator defined during the system design phase. | Implementation<br>EM (B.4.1)      |
|   |                 | Sector common vertex point –<br>confidence {x, y, z} (A.2.207)         | 0   |                                   |
|   |                 | Number of valid vertices<br>(A.2.122)                                  | M   | Optimise LL<br>(B.1.1)            |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid vertices (A.2.122) |   |                                   |
|   Free space<br>area vertex   | M               | Vertex point {x, y, z}<br>(A.2.105)                                    | {M, M, O}   |                                   |
|   |                 | Vertex point {x, y, z} –<br>error (A.2.106)                            | {M, M, O}   | Implementation<br>EM (B.4.1)      |
|   |                 | Vertex point – confidence<br>{x, y, z} (A.2.107)                       | 0   |                                   |
|    Free<br>space area vertex<br>limitation | M               | Limitation reason<br>(A.2.208)   | M   | Default<br>enumerators<br>(B.1.7) |
|   |                 | Limitation moving<br>probability (A.2.209)                             | C   |                                   |
|   |                 | Limitation {radial extent,<br>height} (A.2.210)                        | 0   |                                   |



| LSG | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O | Option                    |
|-----|-----------------|--|--------------------|---------------------------|
|     |                 | Limitation {radial extent, height} – error (A.2.211)     | 0                  | Implementation EM (B.4.1) |
|     |                 | Limitation– confidence {radial extent, height} (A.2.212) | 0                  |                           |
|     |                 | Clearance {height} (A.2.213)                             | 0                  |                           |
|     |                 | Clearance {height} – error (A.2.214)                     | 0                  | Implementation EM (B.4.1) |
|     |                 | Clearance – confidence {height} (A.2.215)                | 0                  |                           |
|     |                 | Object ID reference – object level (A.2.179)             | 0                  |                           |



**Key**


- 1 limitation edge
- 2 free space area shape – signal “Free space type” (A.2.201) := enumerator “FST\_PerceivedLimitedByStaticAndDynamic”
- ↑ limitation {height} information of the height of the limitation

**Figure 19 — Example for free space area objects**

### 7.6.1 Free space area object header

Table 21 defines the interface header in subclause “Free space area object header” (7.6.1) and the changes due to the adaptation in comparison to the generic interface header in 7.2.1.

**Table 21 — Specific signal grouping: Free space area object header**

| Signal   | RL signal | Option  |
|--|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1)   | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)   | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)  | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|  Sensor ID (A.1.4) | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
| Time stamp – prediction (A.1.5.1)  | Mandatory |   |
| Cycle counter (A.1.6.1)  | Optional  | Redundancy (B.1.2)<br>Signal: Time stamp – prediction (A.1.5.1)   |
| Interface cycle time (A.1.7)   | Optional  |   |
| Interface cycle time – variation (A.1.8)   | Optional  |   |
| Data qualifier (A.1.9)   | Mandatory | Default enumerators (B.1.7)   |
| Recognised free space areas – capability (A.1.10.8)  | Optional  |   |
| Recognised free space areas – status (A.1.11.8)  | Optional  | Redundancy (B.1.2)<br>Signal: Recognised free space areas – capability (A.1.10.8)<br>Default enumerators (B.1.7)              |
| Number of valid free space areas (A.1.12.8)  | Mandatory | Optimise LL (B.1.1)   |

The following LSGs, which normally shall be provided by the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor cluster (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

### 7.6.2 Free space area object entity

Each free space area object describes a recognised free space area and, additionally, its limitation. It consists of several LSGs.

A free space area and, additionally, its limitation is described by the following LSGs for free space area.

- **Status:** the status describes the general information of a tracked free space area or limitation. This information is based on history of the object tracking. Table 22 defines the signal grouping “Free space area object entity status”. It defines this status LSG of the free space area and redefines the signal grouping “Generic object level entity status” (see Table 8).

- **Information:** the information describes the free space type and the geometric specification method.
- **Road surface:** the road surface describes the surface of the road inside the defined shape (only for free space area shape and not for the limitation).
- **Shape:** the shape specifies the shape's properties of the defined sector or the closed polyline.
- **Vertex:** the vertex describes the geometric free space area shape or the boundary.
- **Vertex limitation:** the vertex limitation (as reference of an object entity {see signal "Object ID reference – object level" (A.2.179)}) describes the limitation at the geometric boundary of the vertex of the free space area, for example, of a limitation or an underdriveable structure.

**Table 22 — Specific signal grouping: Free space area object entity status**

| Signal   | RL signal | Option   |
|--|-----------|--|
| Existence probability – object level (A.2.1)                         | Mandatory |  |
| Object ID (A.2.2)  | Mandatory | Alternative A2I (B.3.2)  |
| Object grouping ID (A.2.3)   | Optional  |  |
| Age (A.2.4)  | Mandatory |  |
| Number of valid observations – object level (A.2.5)                  | Optional  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – object level (A.2.6)<br>* n recent messages of the interface [see "Time stamp reference – object level" (A.2.6)] |
| <input type="checkbox"/> Time stamp reference – object level (A.2.6) | Mandatory | Alternative VRO (B.1.3)  |
| <input type="checkbox"/> Observation status – object level (A.2.7)   | Mandatory | Redundancy (B.1.2)<br>Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |
| Track quality (A.2.8)  | Optional  |  |
| Measurement status – object level (A.2.9)                            | Mandatory | Default enumerators (B.1.7)  |

## 8 Feature level

### 8.1 General

The feature level interfaces (FLI) originate from several sensors of a sensor cluster. Therefore, at feature level, the term sensor cluster is always used, even if a single sensor is serving the interface. The FLIs provide recognised features, where recognition is based on multiple sensors of the sensor cluster, a small amount of historical data or no historical data at all. The sensor cluster refers its recognised features to one of two predefined vehicle coordinate systems [see signal "Vehicle coordinate system type – header" (A.1.20)]. The feature properties depend on the sensor technology of the sensor cluster. To bring recognised features in relation to entities of other interface levels, each recognised feature has a unique signal's "Feature ID" (A.3.2) value. The uniqueness of the signal's "Feature ID" (A.3.2) value is only guaranteed within the sensor cluster's interfaces.

Wrong link-relations between detections or objects and features shall be prevented. Specifically, the reuse of the signal's "Feature ID" (A.3.2) values shall not lead to a misinterpretation of the link-relation between entities of the sensor cluster's interfaces.

On feature level the following interfaces are available:

- Camera feature interface (8.3);
- Ultrasonic feature interface (8.4).

There is no FLI for radar or lidar sensor clusters defined.

## 8.2 Generic sensor cluster feature interface

Table 23 defines the generic structure of an FLI.


**Table 23 — Generic sensor feature interface**

| Structure   | Multiplicity | Option                      |
|---|--------------|-----------------------------|
| Generic sensor cluster feature header (8.2.1)   | 1            |                             |
| Generic sensor cluster feature entity (8.2.2), specifically an individual feature list interface entity | Multiple     | Size type:<br>dynamic/fixed |

### 8.2.1 Generic sensor cluster feature header

Table 24 defines the interface header in subclause “Generic sensor cluster feature header” (8.2.1) and the changes due to the adaptation in comparison to the generic interface header in 6.2.

**Table 24 — Specific signal grouping: Generic sensor cluster feature header**

| Signal   | RL signal | Option  |
|--|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1)   | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)   | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)  | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|  Sensor ID (A.1.4) | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO(B.1.3)   |
| Time stamp – prediction (A.1.5.1)  | Mandatory |   |
| Cycle counter (A.1.6.1)  | Optional  | Redundancy (B.1.2)<br>Signal: Time stamp – prediction (A.1.5.1)   |
| Interface cycle time (A.1.7)   | Optional  |   |
| Interface cycle time – variation (A.1.8)   | Optional  |   |
| Data qualifier (A.1.9)   | Mandatory | Default enumerators (B.1.7)   |
| <i>FLI sensor technology specific header extensions</i>  |           |   |
| Recognised features – capability (A.1.10.9)  | Optional  |   |
| Recognised features – status (A.1.11.9)  | Optional  | Redundancy (B.1.2)<br>Signal: Recognised features – capability (A.1.10.9)<br>Default enumerators (B.1.7)                      |
| Number of valid features (A.1.12.9)  | Mandatory | Optimise LL (B.1.1)   |

The following LSGs, which normally shall be provided by the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor cluster (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

## 8.2.2 Generic sensor cluster feature entity

Table 25 defines the generic signal grouping “Generic sensor cluster feature entity status” for each FLI and defines the status of a feature [see also Table 8 (7.2.2) for generic object level entities].

**Table 25 — Specific signal grouping: Generic sensor cluster feature entity status**

| Signal   | RL signal                         | Option  |
|--|-----------------------------------|---|
| Existence probability – feature level (A.3.1)                                    | Mandatory                         |   |
| Feature ID (A.3.2)   | Conditional (B.3.1)<br>Exist: DLI | Alternative A2I (B.3.2)   |
| Object ID reference – feature level (A.3.3)                                      | Conditional (B.3.1)<br>Exist: OLI | Alternative A2I (B.3.2)   |
| Time stamp difference – feature level (A.3.4)                                    | Mandatory                         | Optimise IV (B.1.5)   |
| Number of valid observations – feature level (A.3.5)                             | Optional                          | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – feature level (A.3.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
| <input checked="" type="checkbox"/> Time stamp reference – feature level (A.3.6) | Mandatory                         | Alternative VRO (B.1.3)   |
| <input checked="" type="checkbox"/> Observation status – feature level (A.3.7)   | Mandatory                         | Redundancy (B.1.2)<br>Signal: Feature ID (A.3.2) (if unique over time),<br>Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)   |


Each interface type adds individual properties to the interface entity type. Geometric information of the feature references the vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)]. Features reference objects of the sensor cluster. Likewise, detections from individual sensors of a sensor cluster are referencing features.

## 8.3 Camera feature interface

Table 26 defines the logical structure of the camera feature interface (CFI).

Table 26 — Camera feature interface

| LSG                       | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|---------------------------|-----------------|---|--------------------|--|
| Information:<br>interface | M               | Interface version ID {major, minor, patch} (A.1.1)                          | M                  | Profile: Uniqueness of interface versioning (6.4)  |
|                           |                 | Interface ID (A.1.2)  | O                  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)   |
|                           |                 | Number of valid serving sensors (A.1.3)                                     | M                  | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4)  |
|                           |                 | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors (A.1.3) |                    |  |
|                           |                 | Sensor ID (A.1.4)   | M                  | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)   |
|                           |                 | Time stamp – prediction (A.1.5.1)   | M                  |  |
|                           |                 | Cycle counter (A.1.6.1)   | O                  | Redundancy (B.1.2)<br>Signal: Time stamp – prediction (A.1.5.1)  |
|                           |                 | Interface cycle time (A.1.7)  | O                  |  |
|                           |                 | Interface cycle time – variation (A.1.8)                                    | O                  |  |
|                           |                 | Data qualifier (A.1.9)  | M                  | Default enumerators (B.1.7)  |
|                           |                 | Colour model type (A.1.14)  | M                  | Profile: Colour model for CFI (8.3.3)<br>Alternative VRO (B.1.3)<br>Use enumeration to define colours by defining colour values and the applied colour model for each enumerator.<br>Default enumerators (B.1.7) |

| LSG   | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O      | Option  |
|---|-----------------|--|-------------------------|---|
| <b>Features</b>   | M               | Recognised features – capability (A.1.10.9)  | 0                       |   |
|   |                 | Recognised features – status (A.1.11.9)  | 0                       | Redundancy (B.1.2)<br>Signal: Recognised features – capability (A.1.10.9)<br>Default enumerators (B.1.7)  |
|   |                 | Number of valid features (A.1.12.9)  | M                       | Optimise LL (B.1.1)   |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid features (A.1.12.9)                  |                         |   |
|  <b>Status</b> | M               | Existence probability – feature level (A.3.1)  | M                       |   |
|   |                 | Feature ID (A.3.2)   | C (B.3.1)<br>Exist: DLI | Alternative A2I (B.3.2)   |
|   |                 | Feature grouping ID (A.3.8)  | 0                       |   |
|   |                 | Object ID reference – feature level (A.3.3)  | C (B.3.1)<br>Exist: OLI | Alternative A2I (B.3.2)   |
|   |                 | Time stamp difference – feature level (A.3.4)  | M                       | Optimise IV (B.1.5)   |
|   |                 | Number of valid observations – feature level (A.3.5)                                     | 0                       | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – feature level (A.3.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – feature level (A.3.5) |                         |   |
|   |                 | Time stamp reference – feature level (A.3.6)   | M                       | Alternative VRO (B.1.3)   |
|   |                 | Observation status – feature level (A.3.7)   | M                       | Redundancy (B.1.2)<br>Signal: Feature ID (A.3.2) (if unique over time), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)  |

| LSG   | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option  |
|---|-----------------|---|--------------------|---|
| <input type="checkbox"/> Shape information      | M               | Number of valid shape classifications – feature level (A.3.9)   | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Shape classification type – feature level (A.3.10) |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid shape classifications – feature level (A.3.9)                         |                    |   |
|   |                 | Shape classification type – feature level (A.3.10)  | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|   |                 | Shape classification type – confidence – feature level (A.3.11)   | M                  |   |
| <input type="checkbox"/> Shape colour tone      | M               | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |   |
|   |                 | Colour value – feature level (A.3.12)   | M                  | Profile: Colour model for CFI (8.3.3)   |
|   |                 | Colour tone – confidence – feature level (A.3.13)   | O                  | Profile: Colour model for CFI (8.3.3)   |
| <input type="checkbox"/> Shape points           | M               | Shape type – feature level (A.3.14)   | M                  | Default enumerators (B.1.7)   |
|   |                 | Number of valid shape points – feature level (A.3.15)   | M                  | Optimise LL (B.1.1)   |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid shape points – feature level (A.3.15)                                 |                    |   |
|   |                 | Point existence probability – feature level (A.3.16)  | M                  |   |
|   |                 | Position – feature level {x, y, z} (A.3.17)   | M                  |   |
|   |                 | Position – feature level {x, y, z} – error (A.3.18)   | M                  | Implementation EM (B.4.1)   |
| <input type="checkbox"/> Shape reference points | O               | Number of valid shape reference points – feature level (A.3.19)   | M                  | Optimise LL (B.1.1)   |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid shape reference points – feature level (A.3.19)                       |                    |   |
|   |                 | Point existence probability – feature level (A.3.16)  | M                  |   |
|   |                 | Position – feature level {x, y, z} (A.3.17)   | M                  |   |




| LSG | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option                    |
|-----|-----------------|---|--------------------|---------------------------|
|     |                 | Position – feature level {x, y, z} – error (A.3.18)               | M                  | Implementation EM (B.4.1) |
|     |                 | Shape surface normal {x, y, z} (A.3.20)                           | O                  |                           |
|     |                 | Shape surface normal {x, y, z} – error (A.3.21)                   | O                  | Implementation EM (B.4.1) |
|     |                 | Translation rate {x, y, z} – feature level (A.3.22)               | O                  |                           |
|     |                 | Translation rate {x, y, z} – feature level – error (A.3.23)       | O                  | Implementation EM (B.4.1) |
|     |                 | Rotation rate {yaw, pitch, roll} – feature level (A.3.24)         | O                  |                           |
|     |                 | Rotation rate {yaw, pitch, roll} – error – feature level (A.3.25) | O                  | Implementation EM (B.4.1) |
|     |                 | Scale change – feature level (A.3.26)                             | O                  |                           |
|     |                 | Scale change – feature level – error (A.3.27)                     | O                  | Implementation EM (B.4.1) |

### 8.3.1 Camera feature header

Table 27 defines the interface header in subclause “Camera feature header” (8.3.1) and the changes due to the adaptation in comparison to the generic interface header in 8.2.1. The header of the CFI can contain a list of valid camera feature entities (see 8.3.2).

**Table 27 — Specific signal grouping: Camera feature header**

| Signal  | RL signal | Option  |
|---|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1)  | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)  | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)   | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|  Sensor ID (A.1.4) | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
| Time stamp – prediction (A.1.5.1)   | Mandatory |   |
| Cycle counter (A.1.6.1)   | Optional  | Redundancy (B.1.2)<br>Signal: Time stamp – prediction (A.1.5.1)   |
| Interface cycle time (A.1.7)  | Optional  |   |
| Interface cycle time – variation (A.1.8)  | Optional  |   |
| Data qualifier (A.1.9)  | Mandatory | Default enumerators (B.1.7)   |

|   |           |  |
|---|-----------|--|
| Colour model type (A.1.14)                  | Mandatory | Profile: Colour model for CFI (8.3.3)<br>Alternative VRO (B.1.3)<br>Use enumeration to define colours by defining colour values and the applied colour model for each enumerator.<br>Default enumerators (B.1.7) |
| Recognised features – capability (A.1.10.9) | Optional  |  |
| Recognised features – status (A.1.11.9)     | Optional  | Redundancy (B.1.2)<br>Signal: Recognised features – capability (A.1.10.9)<br>Default enumerators (B.1.7)   |
| Number of valid features (A.1.12.9)         | Mandatory | Optimise LL (B.1.1)  |

The following LSGs, which normally shall be provided by the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor cluster (see B.3.3):

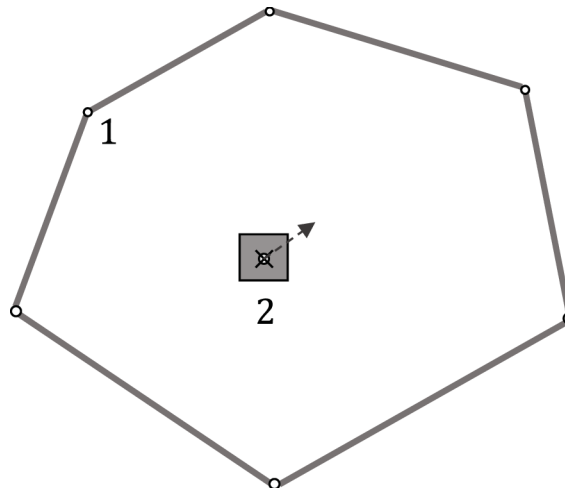
- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

### 8.3.2 Camera feature entity

For the camera, features are understood as shapes extracted from camera images. The shapes describe regions with the same semantic content (see Figure 20).

Each feature of a camera sensor cluster describes a recognised feature and consists of several LSGs.

- **Status:** the status describes general information of the feature and provides an additional grouping ID to group interconnected camera features. This information is based on basic feature history information. Table 28 defines the signal grouping “Camera feature entity status”. It defines this status LSG of the feature and redefines the signal grouping “Generic sensor cluster feature entity status” (see Table 25).
- **Shape information:** the shape information describes the attributes of the tracked shape.
- **Shape colour tone:** the colour tone describes visual properties of the shape.
- **Shape points:** the shape points describe the geometric vertices of the hull for the feature shape.
- **Shape reference points:** the shape reference points are small distinctive trackable segments which are part of the feature’s shape.

**Key**

- 1 shape with shape points
- 2 shape reference point with approximated tangential plane and orientation normal

**Figure 20 — Example for a camera feature shape****Table 28 — Specific signal grouping: Camera feature entity status**

| Signal  | RL signal                         | Option  |
|---|-----------------------------------|---|
| Existence probability – feature level (A.3.1)                         | Mandatory                         |   |
| Feature ID (A.3.2)  | Conditional (B.3.1)<br>Exist: DLI | Alternative A2I (B.3.2)   |
| Feature grouping ID (A.3.8)   | Optional                          |   |
| Object ID reference – feature level (A.3.3)                           | Conditional (B.3.1)<br>Exist: OLI | Alternative A2I (B.3.2)   |
| Time stamp difference – feature level (A.3.4)                         | Mandatory                         | Optimise IV (B.1.5)   |
| Number of valid observations – feature level (A.3.5)                  | Optional                          | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – feature level (A.3.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
| <input type="checkbox"/> Time stamp reference – feature level (A.3.6) | Mandatory                         | Alternative VRO (B.1.3)   |
| <input type="checkbox"/> Observation status – feature level (A.3.7)   | Mandatory                         | Redundancy (B.1.2)<br>Signal: Feature ID (A.3.2) (if unique over time),<br>Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)   |

**8.3.3 Profile: Colour model for CFI**

The signal “Colour model type” (A.1.14) defines the colour model for the interface. Depending on the colour model, each colour is described by a fixed number of colour values [see signal “Colour value –

feature level" (A.3.12)] to define the colour tone. The confidence of the colour tone is provided by the signal "Colour tone – confidence – feature level" (A.3.13).

The interface provides colour tones for features.

#### 8.4 Ultrasonic feature interface

Table 29 defines the logical structure of the ultrasonic feature interface (UFI).

**Table 29 — Ultrasonic feature interface**

| LSG                       | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option  |
|---------------------------|-----------------|---|--------------------|---|
| Information:<br>interface | M               | Interface version ID {major, minor, patch} (A.1.1)                          | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)  |
|                           |                 | Interface ID (A.1.2)  | O                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Default<br>enumerators<br>(B.1.7)   |
|                           |                 | Number of valid serving sensors (A.1.3)                                     | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Optimise LL<br>(B.1.1)<br>Alternative UTL<br>(B.1.6)<br>Key: Sensor ID<br>(A.1.4) |
|                           |                 | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors (A.1.3) |                    |   |
|                           |                 | Sensor ID (A.1.4)   | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Alternative VRO<br>(B.1.3)  |
|                           |                 | Time stamp – prediction (A.1.5.1)   | M                  |   |
|                           |                 | Cycle counter (A.1.6.1)   | O                  | Redundancy<br>(B.1.2)<br>Signal: Time<br>stamp –<br>prediction<br>(A.1.5.1)   |
|                           |                 | Interface cycle time (A.1.7)  | O                  |   |
|                           |                 | Interface cycle time – variation (A.1.8)                                    | O                  |   |

| LSG      | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O      | Option  |
|----------|-----------------|--|-------------------------|---|
|          |                 | Data qualifier (A.1.9)   | M                       | Default enumerators (B.1.7)   |
| Features | M               | Recognised features – capability (A.1.10.9)  | 0                       |   |
|          |                 | Recognised features – status (A.1.11.9)  | 0                       | Redundancy (B.1.2)<br>Signal:<br>Recognised features – capability (A.1.10.9)<br>Default enumerators (B.1.7)   |
|          |                 | Number of valid features (A.1.12.9)  | M                       | Optimise LL (B.1.1)   |
|          |                 | Size type: dynamic/fixed<br>Size #: Number of valid features (A.1.12.9)                  |                         |   |
| ☐ Status | M               | Existence probability – feature level (A.3.1)  | M                       |   |
|          |                 | Feature ID (A.3.2)   | C (B.3.1)<br>Exist: DLI | Alternative A2I (B.3.2)   |
|          |                 | Object ID reference – feature level (A.3.3)  | C (B.3.1)<br>Exist: OLI | Alternative A2I (B.3.2)   |
|          |                 | Time stamp difference – feature level (A.3.4)  | M                       | Optimise IV (B.1.5)   |
|          |                 | Number of valid observations – feature level (A.3.5)                                     | 0                       | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – feature level (A.3.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|          |                 | Size type: dynamic/fixed<br>Size #: Number of valid observations – feature level (A.3.5) |                         |   |
|          |                 | Time stamp reference – feature level (A.3.6)   | M                       | Alternative VRO (B.1.3)   |


| LSG  | RL LSG<br>M/C/O | Signal  |  | RL signal<br>M/C/O | Option   |
|--|-----------------|---|--|--------------------|--|
|  |                 |   | Observation status – feature level (A.3.7)                   | M                  | Redundancy (B.1.2)<br>Signal: Feature ID (A.3.2) (if unique over time), Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7) |
| <input type="checkbox"/> Segment information | M               |   | Number of valid ultrasonic feature classifications (A.3.28)  | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Ultrasonic feature classification type (A.3.29)                                   |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid ultrasonic feature classifications (A.3.28) |  |                    |  |
|  |                 |   | Ultrasonic feature classification type (A.3.29)              | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)   |
|  |                 |   | Ultrasonic feature classification type – confidence (A.3.30) | M                  |  |
| <input type="checkbox"/> Segment points      | M               |   | Number of valid points (A.3.31)                              | M                  | Optimise LL (B.1.1)  |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid points (A.3.31)                             |  |                    |  |
|  |                 |   | Position – feature level {x, y, z} (A.3.17)                  | {M, M, O}          |  |
|  |                 |   | Position – feature level {x, y, z} – error (A.3.18)          | {M, M, O}          | Implementation EM (B.4.1)  |
|  |                 |   | Orientation – feature level {pitch} (A.3.32)                 | O                  |  |
|  |                 |   | Orientation – feature level {pitch} – error (A.3.33)         | O                  | Implementation EM (B.4.1)  |
|  |                 |   | Extent {height} – feature level (A.3.34)                     | O                  |  |
|  |                 |   | Extent {height} – feature level – error (A.3.35)             | O                  | Implementation EM (B.4.1)  |
|  |                 |   | Velocity {x, y} – feature level (A.3.36)                     | O                  |  |

| LSG | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O | Option                      |
|-----|-----------------|--|--------------------|-----------------------------|
|     |                 | Velocity {x, y} – feature level – error (A.3.37) | 0                  | Implementation EM (B.4.1)   |
|     |                 | Trilateration status (A.3.38)                    | M                  | Default enumerators (B.1.7) |
|     |                 | Measurement status – feature level (A.3.39)      | 0                  | Default enumerators (B.1.7) |

#### 8.4.1 Ultrasonic feature header

Table 30 defines the interface header in subclause “Ultrasonic feature header” (8.4.1) and the changes due to the adaptation in comparison to the generic interface header in 8.2.1. The header of the UFI can contain a list of valid ultrasonic feature entities (see 8.4.2).

**Table 30 — Specific signal grouping: Ultrasonic feature header**

| Signal  | RL signal | Option  |
|---|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1)  | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)  | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)   | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|  Sensor ID (A.1.4) | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
| Time stamp – prediction (A.1.5.1)   | Mandatory |   |
| Cycle counter (A.1.6.1)   | Optional  | Redundancy (B.1.2)<br>Signal: Time stamp – prediction (A.1.5.1)   |
| Interface cycle time (A.1.7)  | Optional  |   |
| Interface cycle time – variation (A.1.8)  | Optional  |   |
| Data qualifier (A.1.9)  | Mandatory | Default enumerators (B.1.7)   |
| Recognised features – capability (A.1.10.9)   | Optional  |   |
| Recognised features – status (A.1.11.9)   | Optional  | Redundancy (B.1.2)<br>Signal: Recognised features – capability (A.1.10.9)<br>Default enumerators (B.1.7)                      |
| Number of valid features (A.1.12.9)   | Mandatory | Optimise LL (B.1.1)   |

The following LSGs, which normally shall be provided by the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor cluster (see B.3.3):



- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

#### 8.4.2 Ultrasonic feature entity

Each feature of an ultrasonic sensor cluster describes a recognised feature and consists of several LSGs.

- **Status:** the status describes general information of the feature. This information is based on basic feature history information. Table 31 defines the signal grouping “Ultrasonic feature entity status”. It defines this status LSG of the feature and redefines the signal grouping “Generic sensor cluster feature entity status” (see Table 25).
- **Segment information:** the segment information describes the attributes of the tracked segment.
- **Segment points:** the segment points describe the geometric vertices of the hull for the feature segment.

**Table 31 — Specific signal grouping: Ultrasonic feature entity status**

| Signal  | RL signal                         | Option  |
|---|-----------------------------------|---|
| Existence probability – feature level (A.3.1)   | Mandatory                         |   |
| Feature ID (A.3.2)  | Conditional (B.3.1)<br>Exist: DLI | Alternative A2I (B.3.2)   |
| Object ID reference – feature level (A.3.3)   | Conditional (B.3.1)<br>Exist: OLI | Alternative A2I (B.3.2)   |
| Time stamp difference – feature level (A.3.4)   | Mandatory                         | Optimise IV (B.1.5)   |
| Number of valid observations – feature level (A.3.5)  | Optional                          | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Time stamp reference – feature level (A.3.6)<br>* n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)] |
|  Time stamp reference – feature level (A.3.6) | Mandatory                         | Alternative VRO (B.1.3)   |
|  Observation status – feature level (A.3.7)   | Mandatory                         | Redundancy (B.1.2)<br>Signal: Feature ID (A.3.2) (if unique over time),<br>Time stamp – prediction (A.1.5.1)<br>Default enumerators (B.1.7)   |

## 9 Detection level

### 9.1 General

Normally the detection level interfaces (DLI) originate from one sensing element. Therefore, at detection level, the term sensor is always used, even if, for example, a combination of emitting as well as



sensing elements are serving the interface. The DLIs provide recognised detections. The detections may be based on limited historical data and are rated with individual existence probability values. Detections are defined in the individual sensor coordinate system of the sensor's sensing element. The properties of the detections depend on the sensor technology. The uniqueness of the signal's "Feature ID" (A.3.2) values as well as the signal's "Object ID" (A.2.2) values is only guaranteed within the sensor cluster's interfaces of the sensor. To bring recognised detections in relation to entities of other interface levels, each recognised detections has an optional, unique signal's "Detection ID" (A.4.2) value. The uniqueness of the signal's "Detection ID" (A.4.2) value is only guaranteed within the sensor cluster's interfaces.

Wrong link-relations between features or objects and detections shall be prevented. Specifically, the reuse of the signal's "Detection ID" (A.4.2) values shall not lead to a misinterpretation of the link-relation between entities of the sensor cluster's interfaces.

On detection level the following interfaces are available:

- Radar detection interface (9.3);
- Lidar detection interface (9.4);
- Camera detection interface (9.5);
- Ultrasonic detection interface (9.6).

## 9.2 Generic sensor detection interface

Table 32 defines the generic structure of a DLI.

**Table 32 — Generic sensor detection interface**


| Structure  | Multiplicity | Option                   |
|--|--------------|--------------------------|
| Generic sensor detections header (9.2.1)   | 1            |                          |
| Generic sensor detections entity (9.2.2), specifically an individual detection list interface entity | Multiple     | Size type: dynamic/fixed |

### 9.2.1 Generic sensor detections header

Table 33 defines the interface header in subclause "Generic sensor detections header" (9.2.1) and the changes due to the adaptation in comparison to the generic interface header in 6.2.

**Table 33 — Specific signal grouping: Generic sensor detections header**

| Signal   | RL signal | Option  |
|--|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1) | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)                               | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)            | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |

|  |           |   |
|--|-----------|---|
|  Sensor ID (A.1.4) | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)                                |
| Time stamp – measurement (A.1.5.2)   | Mandatory |   |
| Cycle counter (A.1.6.1)  | Optional  | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)  |
| Interface cycle time (A.1.7)   | Optional  |   |
| Interface cycle time – variation (A.1.8)   | Optional  |   |
| Data qualifier (A.1.9)   | Mandatory | Default enumerators (B.1.7)   |
| DLI sensor technology specific header extensions   |           |   |
| Recognised detections – capability (A.1.10.10)   | Optional  |   |
| Recognised detections – status (A.1.11.10)   | Optional  | Redundancy (B.1.2)<br>Signal: Recognised detections – capability (A.1.10.10)<br>Default enumerators (B.1.7) |
| Number of valid detections (A.1.12.10)   | Mandatory | Optimise LL (B.1.1)<br>Optimise AER (B.1.4)   |

The following LSGs, which normally shall be provided by the sensor in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SPI; LSG “Information: sensor pose” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

### 9.2.2 Generic sensor detections entity

Table 34 defines the sensor grouping “Generic sensor detections entity status” for each DLI and defines the status of the detection [see also Table 8 (7.2.2) for generic object level entities and Table 25 (8.2.2) for generic sensor cluster feature entities].

**Table 34 — Specific signal grouping: Generic sensor detections entity status**

| Signal  | RL signal                         | Option                  |
|---|-----------------------------------|-------------------------|
| Existence probability – detection level (A.4.1) | Mandatory                         |                         |
| Detection ID (A.4.2)                            | Optional                          | Alternative A2I (B.3.2) |
| Object ID reference – detection level (A.4.3)   | Conditional (B.3.1)<br>Exist: OLI | Alternative A2I (B.3.2) |
| Feature ID reference (A.4.4)                    | Conditional (B.3.1)<br>Exist: FLI | Alternative A2I (B.3.2) |
| Time stamp difference – detection level (A.4.5) | Mandatory                         | Optimise IV (B.1.5)     |

Each interface type adds individual properties to the interface entity type. Detections are described in the sensor coordinate system. Detections reference entities of the OLI and FLI of the sensor or the sensor cluster of the sensor.



### 9.3 Radar detection interface

Table 35 defines the logical structure of the radar detection interface (RDI).

**Table 35 — Radar detection interface**

| LSG                              | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O                         | Option  |
|----------------------------------|-----------------|---|--|---|
| Information:<br>interface        | M               | Interface version ID {major, minor, patch} (A.1.1)                          | M  | Profile: Uniqueness of interface versioning (6.4)   |
|                                  |                 | Interface ID (A.1.2)  | O  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
|                                  |                 | Number of valid serving sensors (A.1.3)                                     | M  | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|                                  |                 | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors (A.1.3) |  |   |
|                                  |                 | Sensor ID (A.1.4)   | M  | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
|                                  |                 | Time stamp – measurement (A.1.5.2)  | M  |   |
|                                  |                 | Cycle counter (A.1.6.1)   | O  | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)  |
|                                  |                 | Interface cycle time (A.1.7)  | O  |   |
|                                  |                 | Interface cycle time – variation (A.1.8)                                    | O  |   |
|                                  |                 | Data qualifier (A.1.9)  | M  | Default enumerators (B.1.7)   |
| Information:<br>ambiguity domain | M               | Radial velocity ambiguity domain {begin, end} (A.1.15)                      | C (B.2.1)<br>Relevant: technology specific | Profile: Radar ambiguity (9.3.3)  |
|                                  |                 | Range ambiguity domain {begin, end} (A.1.16)                                | C (B.2.1)<br>Relevant: technology specific | Profile: Radar ambiguity (9.3.3)  |
|                                  |                 | Angle azimuth ambiguity domain {begin, end} (A.1.17)                        | C (B.2.1)<br>Relevant: technology specific | Profile: Radar ambiguity (9.3.3)  |

| LSG                                  | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O                               | Option   |
|--------------------------------------|-----------------|--|--|--|
|                                      |                 | Angle elevation ambiguity domain {begin, end} (A.1.18)                     | C (B.2.1)<br>Relevant:<br>technology<br>specific | Profile: Radar ambiguity (9.3.3)   |
| <b>Detections</b>                    | M               | Recognised detections – capability (A.1.10.10)                             | O  |  |
|                                      |                 | Recognised detections – status (A.1.11.10)                                 | O  | Redundancy (B.1.2)<br>Signal: Recognised detections – capability (A.1.10.10)<br>Default enumerators (B.1.7)  |
|                                      |                 | Number of valid detections (A.1.12.10)                                     | M  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Position {radial distance, azimuth, elevation} (A.4.17)<br>identifiers {azimuth, elevation}<br>Optimise AER (B.1.4) |
|                                      |                 | Size type: dynamic/fixed<br>Size #: Number of valid detections (A.1.12.10) |  |  |
| <input type="checkbox"/> Status      | M               | Existence probability – detection level (A.4.1)                            | M  |  |
|                                      |                 | Detection ID (A.4.2)   | O  | Alternative A2I (B.3.2)  |
|                                      |                 | Object ID reference – detection level (A.4.3)                              | C (B.3.1)<br>Exist: OLI                          | Alternative A2I (B.3.2)  |
|                                      |                 | Time stamp difference – detection level (A.4.5)                            | M  | Optimise IV (B.1.5)  |
| <input type="checkbox"/> Information | M               | Radar cross section (A.4.6)  | M  |  |
|                                      |                 | Radar cross section – error (A.4.7)  | O  | Implementation EM (B.4.1)  |
|                                      |                 | Signal to noise ratio – detection level (A.4.8)                            | M  |  |
|                                      |                 | Signal to noise ratio – detection level – error (A.4.9)                    | O  | Implementation EM (B.4.1)  |
|                                      |                 | Multi target probability (A.4.10)  | O  |  |
|                                      |                 | Ambiguity grouping ID (A.4.11)   | C (B.2.1)<br>Relevant:<br>ambiguity              | Profile: Radar ambiguity (9.3.3)   |
|                                      |                 | Detection ambiguity probability (A.4.12)                                   | C (B.2.1)<br>Relevant:<br>ambiguity              | Profile: Radar ambiguity (9.3.3)   |


| LSG  | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O | Option  |
|--|-----------------|--|--------------------|---|
|  |                 | Free space probability (A.4.13)  | 0                  |   |
|  |                 | Number of valid detection classifications (A.4.14)                                     | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Detection classification type (A.4.15) |
|  |                 | Size type: dynamic/fixed<br>Size #: Number of valid detection classifications (A.4.14) |                    |   |
|  |                 | Detection classification type (A.4.15)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|  |                 | Detection classification type – confidence (A.4.16)                                    | M                  |   |
|  Position   | M               | Position {radial distance, azimuth, elevation} (A.4.17)                                | {M, M, O}          | Alternative VRO (B.1.3)<br>Signal: {azimuth, elevation}                                       |
|  |                 | Position {radial distance, azimuth, elevation} – error (A.4.18)                        | {M, M, O}          | Implementation EM (B.4.1)   |
|  Dynamics | M               | Relative velocity {radial distance} (A.4.19)   | M                  |   |
|  |                 | Relative velocity {radial distance} – error (A.4.20)                                   | 0                  | Implementation EM (B.4.1)   |

### 9.3.1 Radar detections header

Table 36 defines the interface header in subclause “Radar detections header” (9.3.1) and the changes due to the adaptation in comparison to the generic interface header in 9.2.1. The header of the RDI can contain a list of valid radar detection entities (see 9.3.2).

**Table 36 — Specific signal grouping: Radar detections header**

| Signal   | RL signal | Option  |
|--|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1) | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)                               | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)            | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |

| Signal   | RL signal        |  | Option   |
|--|------------------|--|--|
|  Sensor ID (A.1.4) | Mandatory        |  | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)   |
| Time stamp – measurement (A.1.5.2)   | Mandatory        |  |  |
| Cycle counter (A.1.6.1)  | Optional         |  | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)   |
| Interface cycle time (A.1.7)   | Optional         |  |  |
| Interface cycle time – variation (A.1.8)   | Optional         |  |  |
| Data qualifier (A.1.9)   | Mandatory        |  | Default enumerators (B.1.7)  |
| Radial velocity ambiguity domain {begin, end} (A.1.15)   | RL LSG mandatory | Conditional (B.2.1)<br>Relevant: technology specific | Profile: Radar ambiguity (9.3.3)   |
| Range ambiguity domain {begin, end} (A.1.16)   |                  | Conditional (B.2.1)<br>Relevant: technology specific | Profile: Radar ambiguity (9.3.3)   |
| Angle azimuth ambiguity domain {begin, end} (A.1.17)   |                  | Conditional (B.2.1)<br>Relevant: technology specific | Profile: Radar ambiguity (9.3.3)   |
| Angle elevation ambiguity domain {begin, end} (A.1.18)   |                  | Conditional (B.2.1)<br>Relevant: technology specific | Profile: Radar ambiguity (9.3.3)   |
| Recognised detections – capability (A.1.10.10)   | Optional         |  |  |
| Recognised detections – status (A.1.11.10)   | Optional         |  | Redundancy (B.1.2)<br>Signal: Recognised detections – capability (A.1.10.10)<br>Default enumerators (B.1.7)  |
| Number of valid detections (A.1.12.10)   | Mandatory        |  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Position {radial distance, azimuth, elevation} (A.4.17)<br>identifiers {azimuth, elevation}<br>Optimise AER (B.1.4) |

The following LSGs, which normally shall be provided by the sensor in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SPI; LSG “Information: sensor pose” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

### 9.3.2 Radar detections entity

Each detection of a radar sensor describes a measured detection and consists of several LSGs.

- **Status:** the status describes general information of the detection. It is based on current information and does not include historical information. Table 37 defines the signal grouping “Radar detections entity status”. It defines this status LSG of the detection and redefines the signal grouping “Generic sensor detections entity status” (see Table 34). There is no radar feature interface defined and therefore radar detections cannot reference radar features.
- **Information:** the information describes the radar specific information of the detection.
- **Position:** the position describes the position information of the detection in the sensor coordinate system of the detection.
- **Dynamics:** the dynamics describe the dynamic information of the detection in the sensor coordinate system of the detection.

**Table 37 — Specific signal grouping: Radar detections entity status**

| Signal  | RL signal                         | Option                  |
|---|-----------------------------------|-------------------------|
| Existence probability – detection level (A.4.1) | Mandatory                         |                         |
| Detection ID (A.4.2)                            | Optional                          | Alternative A2I (B.3.2) |
| Object ID reference – detection level (A.4.3)   | Conditional (B.3.1)<br>Exist: OLI | Alternative A2I (B.3.2) |
| Time stamp difference – detection level (A.4.5) | Mandatory                         | Optimise IV (B.1.5)     |

### 9.3.3 Profile: Radar ambiguity

The signal “Ambiguity grouping ID” (A.4.11) and the signal “Detection ambiguity probability” (A.4.12) are mandatory, if the radar sensor technology has one or more technology depending ambiguities. The signals corresponding to the following domains of the signal are mandatory, if the sensor technology has the specific ambiguity: ambiguity corresponding to signal “Radial velocity ambiguity domain {begin, end}” (A.1.15), signal “Range ambiguity domain {begin, end}” (A.1.16), signal “Angle azimuth ambiguity domain {begin, end}” (A.1.17) and/or signal “Angle elevation ambiguity domain {begin, end}” (A.1.18).

### 9.4 Lidar detection interface

Table 38 defines the logical structure of the lidar detection interface (LDI).

**Table 38 — Lidar detection interface**

| LSG                       | RL LSG<br>M/C/O | Signal   | RL Signal<br>M/C/O | Option  |
|---------------------------|-----------------|--|--------------------|---|
| Information:<br>interface | M               | Interface version ID {major, minor, patch} (A.1.1) | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)                                      |
|                           |                 | Interface ID (A.1.2)                               | O                  | Profile:<br>Uniqueness of<br>interface<br>versioning (6.4)<br>Default<br>enumerators<br>(B.1.7) |

| LSG               | RL LSG<br>M/C/O | Signal  | RL Signal<br>M/C/O | Option   |
|-------------------|-----------------|---|--------------------|--|
|                   |                 | Number of valid serving sensors (A.1.3)                                     | M                  | Profile:<br>Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|                   |                 | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors (A.1.3) |                    |  |
|                   |                 | Sensor ID (A.1.4)   | M                  | Profile:<br>Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
|                   |                 | Time stamp – measurement (A.1.5.2)  | M                  |  |
|                   |                 | Cycle counter (A.1.6.1)   | O                  | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)   |
|                   |                 | Interface cycle time (A.1.7)  | O                  |  |
|                   |                 | Interface cycle time – variation (A.1.8)                                    | O                  |  |
|                   |                 | Data qualifier (A.1.9)  | M                  | Default enumerators (B.1.7)  |
| <b>Detections</b> | M               | Recognised detections – capability (A.1.10.10)                              | O                  |  |
|                   |                 | Recognised detections – status (A.1.11.10)                                  | O                  | Redundancy (B.1.2)<br>Signal: Recognised detections – capability (A.1.10.10)<br>Default enumerators (B.1.7)                      |



| LSG           | RL LSG<br>M/C/O | Signal   | RL Signal<br>M/C/O      | Option   |
|---------------|-----------------|--|-------------------------|--|
|               |                 | Number of valid detections (A.1.12.10)   | M                       | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Position {radial distance, azimuth, elevation} (A.4.17)<br>Identifiers {azimuth, elevation}<br>Optimise AER (B.1.4) |
|               |                 | Size type: dynamic/fixed<br>Size #: Number of valid detections (A.1.12.10)             |                         |  |
| ☐ Status      | M               | Existence probability – detection level (A.4.1)  | M                       |  |
|               |                 | Detection ID (A.4.2)   | O                       | Alternative A2I (B.3.2)  |
|               |                 | Object ID reference – detection level (A.4.3)  | C (B.3.1)<br>Exist: OLI | Alternative A2I (B.3.2)  |
|               |                 | Time stamp difference – detection level (A.4.5)  | M                       | Optimise IV (B.1.5)  |
| ☐ Information | M               | Reflectivity (A.4.21)  | M                       |  |
|               |                 | Reflectivity – error (A.4.22)  | O                       | Implementation EM (B.4.1)  |
|               |                 | Free space probability (A.4.13)  | O                       |  |
|               |                 | Number of valid detection classifications (A.4.14)                                     | O                       | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Detection classification type (A.4.15)  |
|               |                 | Size type: dynamic/fixed<br>Size #: Number of valid detection classifications (A.4.14) |                         |  |
|               |                 | Detection classification type (A.4.15)   | M                       | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)   |
|               |                 | Detection classification type – confidence (A.4.16)                                    | M                       |  |

| LSG        | RL LSG<br>M/C/O | Signal  | RL Signal<br>M/C/O                                    | Option  |
|------------|-----------------|---|---|---|
| ☐ Position | M               | Position {radial distance, azimuth, elevation} (A.4.17)         | M   | Alternative VRO (B.1.3)<br>Signal: {azimuth, elevation} |
|            |                 | Position {radial distance, azimuth, elevation} – error (A.4.18) | M   | Implementation EM (B.4.1)                               |
|            |                 | Extent {height} – lidar (A.4.23)                                | O   |   |
|            |                 | Extent {height} – lidar – error (A.4.24)                        | C (B.2.2)<br>Signal: Extent {height} – lidar (A.4.23) | Implementation EM (B.4.1)                               |
| ☐ Dynamics | O               | Relative velocity {radial distance} (A.4.19)                    | M   |   |
|            |                 | Relative velocity {radial distance} – error (A.4.20)            | O   | Implementation EM (B.4.1)                               |

#### 9.4.1 Lidar detection header

Table 39 defines the interface header in subclause “Lidar detection header” (9.4.1) and the changes due to the adaptation in comparison to the generic interface header in 9.2.1. The header of the LDI can contain a list of valid lidar detection entities (see 9.4.2).

**Table 39 — Specific signal grouping: Lidar detection header**

| Signal   | RL signal | Option  |
|--|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1) | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)                               | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)            | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
| ☐ Sensor ID (A.1.4)                                | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
| Time stamp – measurement (A.1.5.2)                 | Mandatory |   |
| Cycle counter (A.1.6.1)                            | Optional  | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)  |
| Interface cycle time (A.1.7)                       | Optional  |   |
| Interface cycle time – variation (A.1.8)           | Optional  |   |
| Data qualifier (A.1.9)                             | Mandatory | Default enumerators (B.1.7)   |
| Recognised detections – capability (A.1.10.10)     | Optional  |   |

| Signal                                     | RL signal | Option  |
|--|-----------|---|
| Recognised detections – status (A.1.11.10) | Optional  | Redundancy (B.1.2)<br>Signal: Recognised detections – capability (A.1.10.10)<br>Default enumerators (B.1.7)   |
| Number of valid detections (A.1.12.10)     | Mandatory | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Position {radial distance, azimuth, elevation} (A.4.17) identifiers {azimuth, elevation}<br>Optimise AER (B.1.4) |

The following LSGs, which normally shall be provided by the sensor in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SPI; LSG “Information: sensor pose” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

#### 9.4.2 Lidar detection entity

Each detection of a lidar sensor describes a measured detection and consists of several LSGs.

- **Status:** the status describes general information of the detection. It is based on current information and does not include historical information. Table 40 defines the signal grouping “Lidar detection entity status”. It defines this status LSG of the detection and redefines the signal grouping “Generic sensor detections entity status” (see Table 34). There is no lidar feature interface defined and therefore lidar detections cannot reference lidar features.
- **Information:** the information describes the lidar specific information of the detection.
- **Position:** the position describes the position information of the detection in the sensor coordinate system of the detection.
- **Dynamics:** the dynamics describe the dynamic information of the detection in the sensor coordinate system of the detection.

**Table 40 — Specific signal grouping: Lidar detection entity status**



| Signal  | RL signal                         | Option                  |
|---|-----------------------------------|-------------------------|
| Existence probability – detection level (A.4.1) | Mandatory                         |                         |
| Detection ID (A.4.2)                            | Optional                          | Alternative A2I (B.3.2) |
| Object ID reference – detection level (A.4.3)   | Conditional (B.3.1)<br>Exist: OLI | Alternative A2I (B.3.2) |
| Time stamp difference – detection level (A.4.5) | Mandatory                         | Optimise IV (B.1.5)     |

#### 9.5 Camera detection interface




Table 41 defines the logical structure of the camera detection interface (CDI). Camera detections use shapes and points to define detections.

Table 41 — Camera detection interface

| LSG                       | RL LSG<br>M/C/O | Signal  | RL Signal<br>M/C/O | Option   |
|---------------------------|-----------------|---|--------------------|--|
| Information:<br>interface | M               | Interface version ID {major, minor, patch} (A.1.1)                          | M                  | Profile:<br>Uniqueness of interface versioning (6.4)   |
|                           |                 | Interface ID (A.1.2)  | O                  | Profile:<br>Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
|                           |                 | Number of valid serving sensors (A.1.3)                                     | M                  | Profile:<br>Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|                           |                 | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors (A.1.3) |                    |  |
|                           |                 | Sensor ID (A.1.4)   | M                  | Profile:<br>Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
|                           |                 | Time stamp – measurement (A.1.5.2)  | M                  |  |
|                           |                 | Cycle counter (A.1.6.1)   | O                  | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)   |
|                           |                 | Interface cycle time (A.1.7)  | O                  |  |
|                           |                 | Interface cycle time – variation (A.1.8)                                    | O                  |  |
|                           |                 | Data qualifier (A.1.9)  | M                  | Default enumerators (B.1.7)  |

| LSG  | RL LSG<br>M/C/O                       | Signal   | RL Signal<br>M/C/O      | Option   |
|--|---------------------------------------|--|-------------------------|--|
|  |                                       | Colour model type (A.1.14)   | M                       | Profile: Colour model for CDI (9.5.3)<br>Alternative VRO (B.1.3)<br>Use enumeration to define colours by defining colour values and the applied colour model for each enumerator.<br>Default enumerators (B.1.7) |
| <b>Shapes</b>  | C (B.2.3)<br>multiple<br>entity types | Recognised shapes – capability (A.1.10.11)                             | O                       |  |
|  |                                       | Recognised shapes – status (A.1.11.11)                                 | O                       | Redundancy (B.1.2)<br>Signal: Recognised shapes – capability (A.1.10.11)<br>Default enumerators (B.1.7)  |
|  |                                       | Number of valid shapes (A.1.12.11)                                     | M                       | Optimise LL (B.1.1)  |
|  |                                       | Size type: dynamic/fixed<br>Size #: Number of valid shapes (A.1.12.11) |                         |  |
|  Shape status | M                                     | Existence probability – detection level (A.4.1)                        | M                       |  |
|  |                                       | Detection ID (A.4.2)   | O                       | Alternative A2I (B.3.2)  |
|  |                                       | Object ID reference – detection level (A.4.3)                          | C (B.3.1)<br>Exist: OLI | Alternative A2I (B.3.2)  |
|  |                                       | Feature ID reference (A.4.4)   | C (B.3.1)<br>Exist: FLI | Alternative A2I (B.3.2)  |
|  |                                       | Time stamp difference – detection level (A.4.5)                        | M                       | Optimise IV (B.1.5)  |
|  Shape        | M                                     | Free space probability (A.4.13)  | O                       |  |

| LSG                 | RL LSG<br>M/C/O | Signal  | RL Signal<br>M/C/O   | Option  |
|---------------------|-----------------|---|--|---|
| information         |                 | Number of valid shape classifications – detection level (A.4.25)  | M  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Shape classification type – detection level (A.4.26) |
|                     |                 | Size type: dynamic/fixed<br>Size #: Number of valid shape classifications – detection level (A.4.25)                      |  |   |
|                     |                 | Shape classification type – detection level (A.4.26)  | M  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|                     |                 | Shape classification type – confidence – detection level (A.4.27)   | M  | Default enumerators (B.1.7)   |
|                     |                 | Shape ambiguity grouping ID (A.4.28)  | C (B.2.1)<br>Relevant: ambiguity                                 |   |
| ☐ Shape colour tone | M               | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |  |   |
|                     |                 | Colour value – detection level (A.4.29)   | M  | Profile: Colour model for CDI (9.5.3)   |
|                     |                 | Colour tone – confidence – detection level (A.4.30)   | O  | Profile: Colour model for CDI (9.5.3)   |
| ☐ Shape points      | M               | Shape type – detection level (A.4.31)   | M  | Default enumerators (B.1.7)   |
|                     |                 | Number of valid shape points – detection level (A.4.32)   | M  | Optimise LL (B.1.1)   |
|                     |                 | Size type: dynamic/fixed<br>Size #: Number of valid shape points – detection level (A.4.32)                               |  |   |
|                     |                 | Point existence probability – detection level (A.4.33)  | M  |   |
|                     |                 | Position {radial distance, azimuth, elevation} (A.4.17)   | {O, M, M}  |   |
|                     |                 | Position {radial distance, azimuth, elevation} – error (A.4.18)   | {C (B.2.2)<br>Signal: Position {radial distance} (A.4.17), M, M} | Implementation EM (B.4.1)   |
|                     |                 |   |  |   |

| LSG  | RL LSG<br>M/C/O                    | Signal  | RL Signal<br>M/C/O   | Option  |
|--|------------------------------------|---|--|---|
|  Shape reference points | 0                                  | Number of valid shape reference points – detection level (A.4.34)                                     | M  | Optimise LL (B.1.1)   |
|  |                                    | Size type: dynamic/fixed<br>Size #: Number of valid shape reference points – detection level (A.4.34) |  |   |
|  |                                    | Detection ID reference – detection level (A.4.35)   | M  | <sup>a</sup>  |
| <b>Points</b>  | C (B.2.3)<br>multiple entity types | Recognised detections – capability (A.1.10.10)  | 0  |   |
|  |                                    | Recognised detections – status (A.1.11.10)  | 0  | Redundancy (B.1.2)<br>Signal: Recognised detections – capability (A.1.10.10)<br>Default enumerators (B.1.7) |
|  |                                    | Number of valid detections (A.1.12.10)  | M  | Optimise LL (B.1.1)   |
|  |                                    | Size type: dynamic/fixed<br>Size #: Number of valid detections (A.1.12.10)                            |  |   |
|  Point status         | M                                  | Existence probability – detection level (A.4.1)   | M  |   |
|  |                                    | Detection ID (A.4.2)  | 0  | Alternative A2I (B.3.2)   |
|  |                                    | Object ID reference – detection level (A.4.3)   | C (B.3.1)<br>Exist: OLI  | Alternative A2I (B.3.2)   |
|  |                                    | Feature ID reference (A.4.4)  | C (B.3.1)<br>Exist: FLI  | Alternative A2I (B.3.2)   |
|  |                                    | Time stamp difference – detection level (A.4.5)   | M  | Optimise IV (B.1.5)   |
|  Point information    | M                                  | Position {radial distance, azimuth, elevation} (A.4.17)   | {0, M, M}  |   |
|  |                                    | Position {radial distance, azimuth, elevation} – error (A.4.18)                                       | {C (B.2.2)<br>Signal: Position {radial distance} (A.4.17), M, M} | Implementation EM (B.4.1)   |
|  |                                    | Point existence probability – detection level (A.4.33)  | M  |   |
|  |                                    | Extent {azimuth, elevation} (A.4.36)  | 0  |   |
|  |                                    | Extent {azimuth, elevation} – error (A.4.37)  | 0  | Implementation EM (B.4.1)   |

| LSG  | RL LSG<br>M/C/O | Signal  | RL Signal<br>M/C/O | Option                                |
|--|-----------------|---|--------------------|---------------------------------------|
| <input type="checkbox"/> Point colour tone | M               | Size type: dynamic/fixed<br>Size #<br>* Implicit list length – Optimise LL (B.1.1): depends on Colour model type (A.1.14) |                    |                                       |
|  |                 | Colour value – detection level (A.4.29)   | M                  | Profile: Colour model for CDI (9.5.3) |
|  |                 | Colour tone – confidence – detection level (A.4.30)   | O                  | Profile: Colour model for CDI (9.5.3) |
| <input type="checkbox"/> Point dynamics    | O               | Translation rate {radial distance, azimuth, elevation} – detection level (A.4.38)   | M                  |                                       |
|  |                 | Translation rate {radial distance, azimuth, elevation} – detection level – error (A.4.39)                                 | O                  | Implementation EM (B.4.1)             |
| a Reference only CDI points detections.    |                 |   |                    |                                       |

### 9.5.1 Camera detection header

Table 42 defines the interface header in subclause “Camera detection header” (9.5.1) and the changes due to the adaptation in comparison to the generic interface header in 9.2.1. The header of the CDI can contain a list of valid camera detection entities (see 9.5.2).

**Table 42 — Specific signal grouping: Camera detection header**

| Signal   | RL signal | Option  |
|--|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1) | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)                               | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)            | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
| <input type="checkbox"/> Sensor ID (A.1.4)         | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
| Time stamp – measurement (A.1.5.2)                 | Mandatory |   |
| Cycle counter (A.1.6.1)                            | Optional  | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)  |
| Interface cycle time (A.1.7)                       | Optional  |   |
| Interface cycle time – variation (A.1.8)           | Optional  |   |
| Data qualifier (A.1.9)                             | Mandatory | Default enumerators (B.1.7)   |



| Signal   | RL signal | Option   |
|--|-----------|--|
| Colour model type (A.1.14)                     | Mandatory | Profile: Colour model for CDI (9.5.3)<br>Alternative VRO (B.1.3)<br>Use enumeration to define colours by defining colour values and the applied colour model for each enumerator.<br>Default enumerators (B.1.7) |
| Recognised shapes – capability (A.1.10.11)     | Optional  |  |
| Recognised shapes – status (A.1.11.11)         | Optional  | Redundancy (B.1.2)<br>Signal: Recognised shapes – capability (A.1.10.11)<br>Default enumerators (B.1.7)  |
| Number of valid shapes (A.1.12.11)             | Mandatory | Optimise LL (B.1.1)  |
| Recognised detections – capability (A.1.10.10) | Optional  |  |
| Recognised detections – status (A.1.11.10)     | Optional  | Redundancy (B.1.2)<br>Signal: Recognised detections – capability (A.1.10.10)<br>Default enumerators (B.1.7)  |
| Number of valid detections (A.1.12.10)         | Mandatory | Optimise LL (B.1.1)  |

The following LSGs, which normally shall be provided by the sensor in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SPI; LSG “Information: sensor pose” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

### 9.5.2 Camera detection entity

Each detection of a camera sensor describes a measured shape- or point detection and consists of several LSGs.

A shape is described by the following LSGs for shape.

- **Shape status:** the shape status describes general information of the detection of a shape. It is based on current information and does not include historical information. Table 43 defines the signal grouping “Camera detection entity status”. It defines this status LSG of the detection and redefines the signal grouping “Generic sensor detections entity status” (see Table 34).
- **Shape information:** the information describes the camera specific information of the detection shape.
- **Shape colour tone:** the colour tone describes visual properties of the shape detection.
- **Shape points:** the shape points describe the geometric vertices of the hull for the shape detection in the sensor coordinate system.

- **Shape reference points:** the shape reference points are small distinctive segments which are part of the detection's shape.

A point is described by the following LSGs for point.

- **Point status:** the point status describes general information of the detection of a point. It is based on current information and does not include historical information. Table 43 defines the signal grouping “Camera detection entity status”. It defines this status LSG of the detection and redefines the signal grouping “Generic sensor detections entity status” (see Table 34).
- **Point information:** the information describes the geometric information of the detection point in the sensor coordinate system.
- **Point colour tone:** the colour tone describes visual properties of the point detection.
- **Point dynamics:** the point dynamic describes information of the point detection in the sensor coordinate system.

**Table 43 — Specific signal grouping: Camera detection entity status**

| Signal  | RL signal                         | Option                  |
|---|-----------------------------------|-------------------------|
| Existence probability – detection level (A.4.1) | Mandatory                         |                         |
| Detection ID (A.4.2)                            | Optional                          | Alternative A2I (B.3.2) |
| Object ID reference – detection level (A.4.3)   | Conditional (B.3.1)<br>Exist: OLI | Alternative A2I (B.3.2) |
| Feature ID reference (A.4.4)                    | Conditional (B.3.1)<br>Exist: FLI | Alternative A2I (B.3.2) |
| Time stamp difference – detection level (A.4.5) | Mandatory                         | Optimise IV (B.1.5)     |

### 9.5.3 Profile: Colour model for CDI

The signal “Colour model type” (A.1.14) defines the colour model for the interface. Depending on the colour model, each colour is described by a fixed number of colour values [see signal “Colour value – detection level” (A.4.29)] to define the colour tone. The confidence of the colour tone is provided by the signal “Colour tone – confidence – detection level” (A.4.30).




The interface provides colour tones for shape- and point detections.

### 9.6 Ultrasonic detection interface

Table 44 defines the logical structure of the ultrasonic detection interface (UDI).

Table 44 — Ultrasonic detection interface

| LSG                       | RL LSG<br>M/C/O | Signal  | RL Signal<br>M/C/O | Option  |
|---------------------------|-----------------|---|--------------------|---|
| Information:<br>interface | M               | Interface version ID {major, minor, patch} (A.1.1)                          | M                  | Profile:<br>Uniqueness of interface versioning (6.4)  |
|                           |                 | Interface ID (A.1.2)  | O                  | Profile:<br>Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)   |
|                           |                 | Number of valid serving sensors (A.1.3)                                     | M                  | Profile:<br>Uniqueness of interface versioning (6.4)<br>Profile:<br>Ultrasonic sensor cluster (9.6.3)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|                           |                 | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors (A.1.3) |                    |   |
|                           |                 | Sensor ID (A.1.4)   | M                  | Profile:<br>Uniqueness of interface versioning (6.4)<br>Profile:<br>Ultrasonic sensor cluster (9.6.3)<br>Alternative VRO (B.1.3)  |
|                           |                 | Time stamp – measurement (A.1.5.2)  | M                  |   |
|                           |                 | Cycle counter (A.1.6.1)   | O                  | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)  |
|                           |                 | Interface cycle time (A.1.7)  | O                  |   |
|                           |                 | Interface cycle time – variation (A.1.8)                                    | O                  |   |


| LSG   | RL LSG<br>M/C/O | Signal   | RL Signal<br>M/C/O                                 | Option   |
|---|-----------------|--|--|--|
|   |                 | Data qualifier (A.1.9)   | M  | Default enumerators (B.1.7)  |
| <b>Detections</b>   | M               | Recognised detections – capability (A.1.10.10)                             | O  |  |
|   |                 | Recognised detections – status (A.1.11.10)                                 | O  | Redundancy (B.1.2)<br>Signal:<br>Recognised detections – capability (A.1.10.10)<br>Default enumerators (B.1.7) |
|   |                 | Number of valid detections (A.1.12.10)                                     | M  | Optimise LL (B.1.1)<br>Optimise AER (B.1.4)  |
|   |                 | Size type: dynamic/fixed<br>Size #: Number of valid detections (A.1.12.10) |  |  |
|  Status      | M               | Existence probability – detection level (A.4.1)                            | M  |  |
|   |                 | Detection ID (A.4.2)   | O  | Alternative A2I (B.3.2)  |
|   |                 | Object ID reference – detection level (A.4.3)                              | C (B.3.1)<br>Exist: OLI                            | Alternative A2I (B.3.2)  |
|   |                 | Feature ID reference (A.4.4)   | C (B.3.1)<br>Exist: FLI                            | Alternative A2I (B.3.2)  |
|   |                 | Time stamp difference – detection level (A.4.5)                            | M  | Profile:<br>Ultrasonic sensor cluster (9.6.3)<br>Optimise IV (B.1.5)   |
|  Information | M               | Second sensor ID reference (A.4.40)  | C<br>Profile:<br>Ultrasonic sensor cluster (9.6.3) | Profile:<br>Ultrasonic sensor cluster (9.6.3)  |
|   |                 | Reflectivity (A.4.21)  | O  |  |
|  Position    | M               | Distance (A.4.41)  | M  |  |
|   |                 | Distance – error (A.4.42)  | M  | Implementation EM (B.4.1)  |

| LSG | RL LSG<br>M/C/O | Signal   | RL Signal<br>M/C/O                               | Option                       |
|-----|-----------------|--|--|------------------------------|
|     |                 | Extent {height} – ultrasonic (A.4.43)            | C (B.2.1)<br>Relevant:<br>technology<br>specific |                              |
|     |                 | Extent {height} – ultrasonic – error<br>(A.4.44) | 0  | Implementation<br>EM (B.4.1) |

### 9.6.1 Ultrasonic detection header

Table 45 defines the interface header in subclause “Ultrasonic detection header” (9.6.1) and the changes due to the adaptation in comparison to the generic interface header in 9.2.1. The header of the UDI can contain a list of valid ultrasonic detection entities (see 9.6.2).

**Table 45 — Specific signal grouping: Ultrasonic detection header**

| Signal  | RL signal | Option  |
|---|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1)  | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)  | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)   | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Profile: Ultrasonic sensor cluster (9.6.3)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|  Sensor ID (A.1.4) | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Profile: Ultrasonic sensor cluster (9.6.3)<br>Alternative VRO (B.1.3)  |
| Time stamp – measurement (A.1.5.2)  | Mandatory |   |
| Cycle counter (A.1.6.1)   | Optional  | Redundancy (B.1.2)<br>Signal: Time stamp – measurement (A.1.5.2)  |
| Interface cycle time (A.1.7)  | Optional  |   |
| Interface cycle time – variation (A.1.8)  | Optional  |   |
| Data qualifier (A.1.9)  | Mandatory | Default enumerators (B.1.7)   |
| Recognised detections – capability (A.1.10.10)  | Optional  |   |
| Recognised detections – status (A.1.11.10)  | Optional  | Redundancy (B.1.2)<br>Signal: Recognised detections – capability (A.1.10.10)<br>Default enumerators (B.1.7)   |
| Number of valid detections (A.1.12.10)  | Mandatory | Optimise LL (B.1.1)<br>Optimise AER (B.1.4)   |

The UDI may provide all detections of either the emitting or the sensing ultrasonic sensor. Each detection will provide the corresponding sensor by the signal “Second sensor ID reference” (A.4.40). If the UDI provides detections for a sensor-sub-cluster, the first signal “Sensor ID” (A.1.4) provides the emitting sensor and the following signal “Sensor ID” (A.1.4) provides the sensing sensor. In case of a

concurrent emitting and sensing sensor element the signal “Sensor ID” (A.1.4) provides only one ID for both elements (see 9.6.3).

The following LSGs, which normally shall be provided by the sensor in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SPI; LSG “Information: sensor pose” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

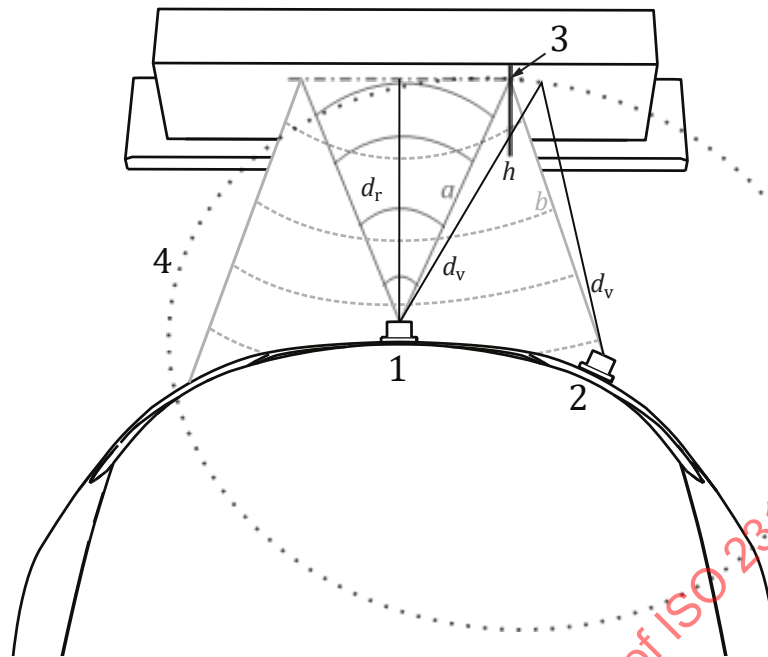
### 9.6.2 Ultrasonic detection entity

Each detection of an ultrasonic sensor describes a measured detection and consists of several LSGs.

- **Status:** the status describes general information of the detection. It is based on current information and does not include historical information. Table 46 defines the signal grouping “Ultrasonic detection entity status”. It defines this status LSG of the detection and redefines the signal grouping “Generic sensor detections entity status” (see Table 34).
- **Information:** The information describes the ultrasonic specific information of the echo (see Figure 21).

**Table 46 — Specific signal grouping: Ultrasonic detection entity status**

| Signal  | RL signal                         | Option                  |
|---|-----------------------------------|-------------------------|
| Existence probability – detection level (A.4.1) | Mandatory                         |                         |
| Detection ID (A.4.2)                            | Optional                          | Alternative A2I (B.3.2) |
| Object ID reference – detection level (A.4.3)   | Conditional (B.3.1)<br>Exist: OLI | Alternative A2I (B.3.2) |
| Feature ID reference (A.4.4)                    | Conditional (B.3.1)<br>Exist: FLI | Alternative A2I (B.3.2) |
| Time stamp difference – detection level (A.4.5) | Mandatory                         | Optimise IV (B.1.5)     |

**Key**

- 1 ultrasonic sender ( $\overrightarrow{S}$ ) – emitting ultrasonic element
- 2 ultrasonic receiver ( $\overrightarrow{R}$ ) – sensing ultrasonic element
- 3 obstacle's reflection point ( $\overrightarrow{O}$ )
- 4 ellipse (theoretically possible points of an obstacle's reflection)
- $a$  distance from the emitting ultrasonic element to the obstacle's reflection point  $\coloneqq |\overrightarrow{O} - \overrightarrow{S}|$
- $b$  distance from the obstacle's reflection point to the sensing ultrasonic element  $\coloneqq |\overrightarrow{R} - \overrightarrow{O}|$
- $d_v$  mean distance of the echo of a reflection between emitting ultrasonic element (see key 1), obstacle's reflection point (see key 3) and sensing ultrasonic element (see key 2)  $\coloneqq (a + b)/2$
- $d_r$  distance of a direct reflection between obstacle and the emitting and sensing ultrasonic element (see key 1)
- $h$  height of the obstacle

**Figure 21 — Example for an ultrasonic sensor cluster with separate emitting and sensing elements on the example of two received echoes of one emitted pulse**

### 9.6.3 Profile: Ultrasonic sensor cluster

An ultrasonic sensor cluster (see Figure 22) may serve this interface in different ways (see Table 47).

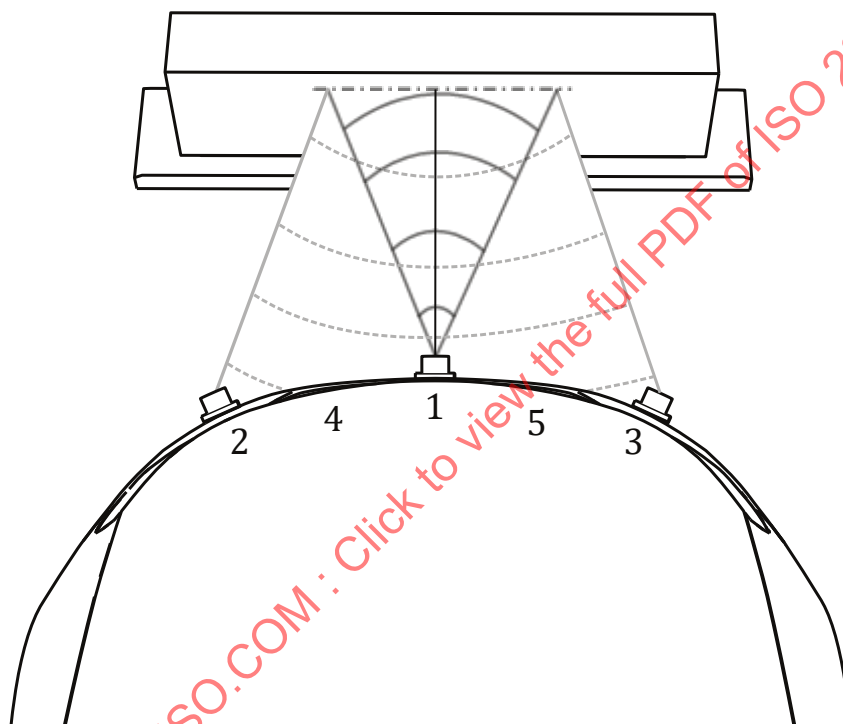
1. Each virtual sensor (which is based on two different real sensors, one for emitting and one for sensing) and real sensor (which is based on one real sensors, which is emitting and sensing its emitted pulse) provide a separate interface. The emitting and sensing sensors are defined in the header of the interface message. Real sensors, which emit and sense their own echo, provide a DLI with the signal "Sensor ID" (A.1.4) of the sensor (one ID). Virtual sensors provide a DLI with the first signal "Sensor ID" (A.1.4) of the emitting sensor and the second of the sensing sensor (two IDs). The RL of the signal "Second sensor ID reference" (A.4.40) is optional and the signal would be redundant.
2. Each real sensor provides a separate interface, including all detections of the virtual sensors. Virtual sensors may serve no interface.
  - a. The signal "Sensor ID" (A.1.4) provides the ID of the emitting sensor. The sensing sensor of the detection is referenced by the signal "Second sensor ID reference" (A.4.40), which is mandatory

for a sensor cluster. The measurement cycle of the sensor cluster's UDIs may not overlap in time.

- b. The signal "Sensor ID" (A.1.4) provides the ID of the sensing sensor. The emitting sensor of the detection is referenced by the signal "Second sensor ID reference" (A.4.40), which is mandatory for a sensor cluster. The measurement cycle of the sensor cluster's UDIs may overlap in time.

The signal "Second sensor ID reference" (A.4.40) is optional for a single ultrasonic sensor.

All emitting and/or sensing ultrasonic sensors may provide the SPI sensor origin point and sensor axis system, in detail the LSG "Information: sensor pose" (see Table 51). Even if the ultrasonic detections are clustered in the interface (see Profile: Ultrasonic sensor cluster (9.6.3) – option 2a or 2b) or the origin point and sensor axis system of a virtual sensor is calculated of the sensor poses of the real sensors' SPIs.



**Key**

- 1 ultrasonic sender and receiver (SR)  
real sensor {SR} origin
- 2 ultrasonic receiver (R1)
- 3 ultrasonic receiver (R2)
- 4 virtual sensor {SR, R1} origin
- 5 virtual sensor {SR, R2} origin

**Figure 22 — Example for an ultrasonic sensor cluster of 3 ultrasonic elements for one echo and the resulting sensor origins**



**Table 47 — Example of UDIs for 3 emitting and sensing elements; one echo – Profile relevant signals**

| Profile: Ultrasonic sensor cluster (9.6.3) – option 1 (9 UDI) | Profile: Ultrasonic sensor cluster (9.6.3) – option 2a (3 UDI)   | Profile: Ultrasonic sensor cluster (9.6.3) – option 2b (3 UDI)  |
|---|--|---|
| UDI {<br>{Sensor ID = SR}<br>Detection {<br>}<br>}            | UDI {<br>{Sensor ID = SR}<br>Detection {<br>Second sensor ID reference = SR<br>Time stamp difference = $dt_{SR SR}$<br>}<br>Detection {<br>Second sensor ID reference = R1<br>Time stamp difference = $dt_{SR R1}$<br>}<br>} | UDI {<br>{Sensor ID = SR}<br>...<br>Detection {<br>Second sensor ID reference = SR<br>Time stamp difference = $dt_{SR} + dt_{SR SR}$<br>}<br>...<br>} |
| UDI {<br>{Sensor ID = {SR, R1}}<br>Detection {<br>}<br>}      | UDI {<br>{Sensor ID = {SR, R1}}<br>Detection {<br>Second sensor ID reference = R2<br>Time stamp difference = $dt_{SR R2}$<br>}<br>}  | UDI {<br>{Sensor ID = R1}<br>...<br>Detection {<br>Second sensor ID reference = SR<br>Time stamp difference = $dt_{SR} + dt_{SR R1}$<br>}<br>...<br>} |
| UDI {<br>{Sensor ID = {SR, R2}}<br>Detection {<br>}<br>}      | ...  | UDI {<br>{Sensor ID = R2}<br>...<br>Detection {<br>Second sensor ID reference = SR<br>Time stamp difference = $dt_{SR} + dt_{SR R2}$<br>}<br>...<br>} |
| ...   |  |   |

## 10 Supportive sensor interfaces

### 10.1 General

Sensors and sensor clusters serve supportive sensor interfaces (SSI). A fusion unit may derive the sensor cluster's SSI's information of the sensors' SSIs of the sensor cluster. Therefore, in this clause for SSI, the term sensor is always used, even if a sensor cluster is serving the interface. The SSIs provide additional, general information of a sensor's or sensor cluster's health and performance. Table 48 provides an overview over the differences and boundaries between the SSIs.

**Table 48 — Brief overview over the differences and boundaries between the SSIs**

| Sensor performance   |   | Sensor health information   |
|--|---|---|
| Impairment on observed field   | Impairment on sensor surface  | Sensor internal   |
| <ul style="list-style-type: none"> <li>— rain, fog</li> <li>— snow</li> <li>— particles (air)</li> <li>— and so forth</li> </ul> | <ul style="list-style-type: none"> <li>— dirt, dust</li> <li>— condensation</li> <li>— scratch</li> <li>— and so forth</li> </ul> | <ul style="list-style-type: none"> <li>— operation</li> <li>— diagnosis</li> <li>— defects</li> <li>— cleaning</li> <li>— position calibration</li> <li>— and so forth</li> </ul> |
| Classification into several measurement ranges<br>Relevant for safety concept  |   | Global information for the complete sensor<br>Misalignment: hardware and software calibration<br>Relevant E/E information   |

The following supportive sensor interfaces are available:

- Sensor performance interface (10.3);
- Sensor health information interface (10.4).

Supportive sensor interface messages (see 10.3 and 10.4) may be required to correctly interpret information from object-, feature- and detection level interface. Data consistency over the sensor's interfaces shall be assured. Supportive sensor interfaces have LSGs which the sensor shall provide. If an SSI is not provided, the SSI relevant LSG will be provided by other interfaces as defined (see B.3.3).

## 10.2 Generic supportive sensor interface

Table 49 defines the generic structure of an SSI.

**Table 49 — Generic supportive sensor interface**

| Structure   | Multiplicity                        | Option                   |
|---|-------------------------------------|--------------------------|
| Generic supportive sensor header (10.2.1)   | 1                                   |                          |
| Generic supportive sensor entity (10.2.2), specifically an individual supportive sensor entity (list) | SPI: multiple as well as<br>SHII: 1 | Size type: dynamic/fixed |

The multiplicity of the individual supportive sensor interface entity list depends on the individual SSI interface.


The geometric information of the SSIs references the vehicle coordinate system (see signal "Vehicle coordinate system type – header" (A.1.20)) as well as the sensor coordinate system.

### 10.2.1 Generic supportive sensor header

Table 50 defines the interface header in subclause "Generic supportive sensor header" (10.2.1) and the changes due to the adaptation in comparison to the generic interface header in 6.2. Each individual SSI may define the signal "Time stamp – <...>" (A.1.5) during the system design phase.

**Table 50 — Specific signal grouping: Generic supportive sensor header**

| Signal   | RL signal | Option  |
|--|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1) | Mandatory | Profile: Uniqueness of interface versioning (6.4) |

|   |           |   |
|---|-----------|---|
| Interface ID (A.1.2)  | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)   | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|  Sensor ID (A.1.4) | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
| Time stamp – <...> (A.1.5)  | Mandatory |   |
| Message counter (A.1.6.2)   | Optional  |   |
| Interface cycle time (A.1.7)  | Optional  |   |
| Interface cycle time – variation (A.1.8)  | Optional  |   |
| Data qualifier (A.1.9)  | Mandatory | Default enumerators (B.1.7)   |
| SSI specific header extensions  |           |   |
| Number of valid <...> (A.1.12)  | Mandatory | Optimise LL (B.1.1)   |

The LSGs, which normally shall be provided by the sensor as well as the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor or sensor cluster (see B.3.3). For each SSI these LSGs are individual.

### 10.2.2 Generic supportive sensor entity

No generic LSG is defined for the interface entity types of all SSIs.

### 10.3 Sensor performance interface

Table 51 defines the logical structure of the sensor performance interface (SPI).

**Table 51 — Sensor performance interface**

| LSG                       | RL LSG<br>M/C/O | Signal  | RL Signal<br>M/C/O | Option  |
|---------------------------|-----------------|---|--------------------|---|
| Information:<br>interface | M               | Interface version ID {major,<br>minor, patch} (A.1.1) | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning of SPIs<br>(10.3.3)                                      |
|                           |                 | Interface ID (A.1.2)                                  | O                  | Profile:<br>Uniqueness of<br>interface<br>versioning of SPIs<br>(10.3.3)<br>Default<br>enumerators<br>(B.1.7) |
|                           |                 | Number of valid serving sensors<br>(A.1.3)            | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning of SPIs<br>(10.3.3)<br>Optimise LL                       |

| LSG   | RL LSG<br>M/C/O   | Signal  | RL Signal<br>M/C/O | Option  |
|---|---|---|--------------------|---|
|   |   |   |                    | (B.1.1)<br>Alternative UTL<br>(B.1.6)<br>Key: Sensor ID<br>(A.1.4)  |
|   |   | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors (A.1.3) |                    |   |
|   |   | Sensor ID (A.1.4)   | M                  | Profile:<br>Uniqueness of<br>interface<br>versioning of SPIs<br>(10.3.3)<br>Alternative VRO<br>(B.1.3)        |
|   |   | Time stamp – <...> (A.1.5)  | M                  |   |
|   |   | Message counter (A.1.6.2)   | O                  |   |
|   |   | Interface cycle time (A.1.7)  | O                  |   |
|   |   | Interface cycle time – variation<br>(A.1.8)                                 | O                  |   |
|   |   | Data qualifier (A.1.9)  | M                  | Default<br>enumerators<br>(B.1.7)   |
|   |   | Interface applicability (A.1.19)  | O                  | Profile:<br>Uniqueness of<br>interface<br>versioning of SPIs<br>(10.3.3)<br>Default<br>enumerators<br>(B.1.7) |
| Information:<br>vehicle<br>coordinate<br>system | M<br>Need of logical signal<br>group (B.3.3)<br>1. SHII Header,<br>2. Redundant on OLI,<br>FLI and DLI Header | Vehicle coordinate system type<br>– header (A.1.20)                         | M                  | Optimise IV<br>(B.1.5)  |
| Information:<br>sensor pose                     | M<br>Need of logical signal<br>group (B.3.3)<br>1. SHII Header,<br>2. DLI Header                              | Sensor origin point {x, y, z} –<br>header (A.1.21)                          | M                  |   |
|   |   | Sensor origin point {x, y, z} –<br>error (A.1.22)                           | O                  | Implementation<br>EM (B.4.1)  |
|   |   | Sensor orientation {yaw, pitch,<br>roll} – header (A.1.23)                  | M                  |   |
|   |   | Sensor orientation {yaw, pitch,<br>roll} – error (A.1.24)                   | O                  | Implementation<br>EM (B.4.1)  |
| Information:<br>sensor                          | O   | Vanishing point {azimuth,<br>elevation} (A.1.25)                            | M                  |   |

| LSG                       | RL LSG<br>M/C/O | Signal  | RL Signal<br>M/C/O | Option   |
|---------------------------|-----------------|---|--------------------|--|
| surrounding               |                 | Vanishing point {azimuth, elevation} – error (A.1.26)                                       | 0                  | Implementation EM (B.4.1)  |
| Segments                  | M               | Number of valid field-of-view segments (A.1.12.12)  | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Segment azimuth – supportive sensor level {begin, end} (A.5.1) and Segment elevation – supportive sensor level {begin, end} (A.5.2) |
|                           |                 | Size type: Dynamic/fixed<br>Size #: Number of valid field-of-view segments (A.1.12.12)      |                    |  |
| ☐ Status                  | M               | Segment azimuth – supportive sensor level {begin, end} (A.5.1)                              | M                  | Alternative VRO (B.1.3)  |
|                           |                 | Segment elevation – supportive sensor level {begin, end} (A.5.2)                            | 0                  | Alternative VRO (B.1.3)  |
|                           |                 | Measurement grid resolution {radial distance, azimuth, elevation} (A.5.3)                   | 0                  |  |
|                           |                 | Beam divergence {azimuth, elevation} (A.5.4)  | 0                  |  |
|                           |                 | Range gain (A.5.5)  | 0                  |  |
|                           |                 | Blockage status (A.5.6)   | M                  | Default enumerators (B.1.7)  |
| ☐ Field-of-view reduction | 0               | Number of valid field-of-view reduction reasons (A.5.7)                                     | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Field-of-view reduction reason type (A.5.8)   |
|                           |                 | Size type: Dynamic/fixed<br>Size #: Number of valid field-of-view reduction reasons (A.5.7) |                    |  |
|                           |                 | Field-of-view reduction reason type (A.5.8)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)   |

| LSG  | RL LSG<br>M/C/O   | Signal   | RL Signal<br>M/C/O                             | Option   |
|--|---|--|--|--|
|  |   | Field-of-view reduction<br>reason type – confidence<br>(A.5.9)                         | M  |  |
| <input type="checkbox"/> Real-world<br>object<br>recognition<br>capabilities | C (B.3.1)<br>Exist: OLI<br>and in scope of signal<br>“Interface<br>applicability” (A.1.19)      | Number of valid<br>recognisable object types<br>(A.5.10)                               | M  | Optimise LL<br>(B.1.1)<br>Alternative UTL<br>(B.1.6)<br>Key: Recognised<br>object type<br>(A.5.11) |
|  |   | Size type: Dynamic/fixed<br>Size #: Number of valid recognisable object types (A.5.10) |  |  |
|  |   | Recognised object type<br>(A.5.11)   | M  | Alternative VRO<br>(B.1.3)<br>Default<br>enumerators<br>(B.1.7)                                    |
|  |   | Detection range radial<br>distance {begin, end}<br>(A.5.12)                            | M  |  |
|  |   | True positive rate<br>(A.5.13)   | O  |  |
|  |   | False positive rate<br>(A.5.14)  | O  |  |
|  |   | Positive predictive value<br>(A.5.15)  | O  |  |
|  |   | Number of valid reference<br>target types (A.5.16)                                     | M  | Optimise LL<br>(B.1.1)   |
| <input type="checkbox"/> Reference<br>target<br>recognition<br>capabilities  | C (B.3.1)<br>Exist: FLI, DLI<br>and in scope of signal<br>“Interface<br>applicability” (A.1.19) | Size type: Dynamic/fixed<br>Size #: Number of valid reference target types (A.5.16)    |  |  |
|  |   | Reference target type<br>(A.5.17)  | C (B.2.1)<br>Relevant:<br>camera               | Default<br>enumerators<br>(B.1.7)  |
|  |   | Radar cross section<br>reference target (A.5.18)                                       | C (B.2.1)<br>Relevant:<br>radar                |  |
|  |   | Reflectivity reference<br>target (A.5.19)  | C (B.2.1)<br>Relevant:<br>lidar,<br>ultrasonic |  |
|  |   | Detection range radial<br>distance {begin, end}<br>(A.5.12)                            | M  |  |
|  |   | True positive rate<br>(A.5.13)   | O  |  |
|  |   | Relative radial velocity   | O  |  |

| LSG | RL LSG<br>M/C/O | Signal   | RL Signal<br>M/C/O | Option                 |
|-----|-----------------|--|--------------------|------------------------|
|     |                 | range {begin, end}<br>(A.5.20)   |                    |                        |
|     |                 | Signal to noise ratio –<br>supportive level (A.5.21)                       | M                  | Optimise IV<br>(B.1.5) |
|     |                 | Spatial separability<br>{radial distance, azimuth,<br>elevation} (A.5.22)  | O                  |                        |
|     |                 | Velocity separability<br>{radial distance, azimuth,<br>elevation} (A.5.23) | O                  |                        |

The signals referring to angles as well as distances in the segments always refer to the sensor coordinate system and its origin. This affects the signals:

- Segment azimuth – supportive sensor level {begin, end} (A.5.1);
- Segment elevation – supportive sensor level {begin, end} (A.5.2);
- Measurement grid resolution {radial distance, azimuth, elevation} (A.5.3);
- Detection range radial distance {begin, end} (A.5.12);
- Relative radial velocity range {begin, end} (A.5.20);
- Spatial separability {radial distance, azimuth, elevation} (A.5.22);
- Velocity separability {radial distance, azimuth, elevation} (A.5.23).


The correlation between vehicle- and sensor coordinate system is determined by the sensor origin point and the sensor orientation of the sensor (LSG “Information: sensor pose” – see Table 51).

### 10.3.1 Sensor performance header

Table 52 defines the interface header in subclause “Sensor performance header” (10.3.1) and the changes due to the adaptation in comparison to the generic interface header in 10.2.1. The header of the SPI can contain a list of valid FOV segment entities (see 10.3.2).

**Table 52 — Specific signal grouping: Sensor performance header**

| Signal   | RL signal | Option   |
|--|-----------|--|
| Interface version ID {major, minor, patch} (A.1.1) | Mandatory | Profile: Uniqueness of interface versioning of SPIs (10.3.3)   |
| Interface ID (A.1.2)                               | Optional  | Profile: Uniqueness of interface versioning of SPIs (10.3.3)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)            | Mandatory | Profile: Uniqueness of interface versioning of SPIs (10.3.3)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |

| Signal  | RL signal  |           | Option   |
|---|--|-----------|--|
|  Sensor ID (A.1.4) | Mandatory  |           | Profile: Uniqueness of interface versioning of SPIs (10.3.3)<br>Alternative VRO(B.1.3)   |
| Time stamp – <...> (A.1.5)  | Mandatory  |           |  |
| Message counter (A.1.6.2)   | Optional   |           |  |
| Interface cycle time (A.1.7)  | Optional   |           |  |
| Interface cycle time – variation (A.1.8)  | Optional   |           |  |
| Data qualifier (A.1.9)  | Mandatory  |           | Default enumerators (B.1.7)  |
| Interface applicability (A.1.19)  | Optional   |           | Profile: Uniqueness of interface versioning of SPIs (10.3.3)<br>Default enumerators (B.1.7)  |
| Vehicle coordinate system type – header (A.1.20)  | RL LSG mandatory<br>Need of logical signal group (B.3.3)<br>1. SHII Header,<br>2. Redundant on OLI, FLI and DLI Header | Mandatory | Optimise IV (B.1.5)  |
| Sensor origin point {x, y, z} – header (A.1.21)   | RL LSG mandatory<br>Need of logical signal group (B.3.3)<br>1. SHII Header,<br>2. DLI Header                           | Mandatory |  |
| Sensor origin point {x, y, z} – error (A.1.22)  |  | Optional  | Implementation EM (B.4.1)  |
| Sensor orientation {yaw, pitch, roll} – header (A.1.23)   |  | Mandatory |  |
| Sensor orientation {yaw, pitch, roll} – error (A.1.24)  |  | Optional  | Implementation EM (B.4.1)  |
| Vanishing point {azimuth, elevation} (A.1.25)   | RL LSG optional  | Mandatory |  |
| Vanishing point {azimuth, elevation} – error (A.1.26)   |  | Optional  | Implementation EM (B.4.1)  |
| Number of valid field-of-view segments (A.1.12.12)  | Mandatory  |           | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Segment azimuth – supportive sensor level {begin, end} (A.5.1) and Segment elevation – supportive sensor level {begin, end} (A.5.2) |

The following LSGs, which normally shall be provided by the sensor as well as the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor or sensor cluster (see B.3.3):

- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).



The following LSGs, which shall be provided by the sensor as well as the sensor cluster in this interface, may be added to another header (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SPI; LSG “Information: sensor pose” (see Table 51).

### 10.3.2 Sensor performance entity

Each sensor performance segment for FOV of a sensor describes the performance in this segment and consists of several LSGs.

- **Status:** the status describes the geometry of the FOV segment.
- **Field-of-view reduction causes:** the FOV reduction causes describe the causes (for example, extrinsic or intrinsic) for an FOV reduction.
- **Real-world object recognition rates:** the real-world object recognition rates describe the estimated recognition rates for real-world recognition of different real-world objects.
- **Reference target recognition rates:** the reference target recognition rates describe the estimated recognition rates for synthetic, well-defined recognition of different synthetic, well-defined objects.

For two valid but geometric overlapping FOV segments (azimuth-, elevation range), the values for the earlier defined segment in the list will always stay valid. So typically, the segments will be defined from the inside to the outside (due to timing reasons, special regions of interest are more important than the general FOV segments and the special regions of interest need to be known to be subtracted from the general FOV segment).

### 10.3.3 Profile: Uniqueness of interface versioning of SPIs

This profile extends the profile of 6.4. A sensor cluster may provide this interface more than once for the sensor. To identify the different interface implementations of the SPI the signal “Interface applicability” (A.1.19) is additionally used to define uniqueness.

## 10.4 Sensor health information interface

The subclause “Sensor health information interface” (10.4) (SHII) provides a qualitative statement of the sensor’s health status. More detailed information of sensor statuses can be provided via other interfaces.

Table 53 defines the logical structure of the sensor health information interface.

**Table 53 — Sensor health information interface**

| LSG                       | RL LSG<br>M/C/O | Signal   | RL Signal<br>M/C/O | Option   |
|---------------------------|-----------------|--|--------------------|--|
| Information:<br>interface | M               | Interface version ID {major, minor, patch} (A.1.1) | M                  | Profile: Uniqueness of interface versioning (6.4)                                |
|                           |                 | Interface ID (A.1.2)                               | O                  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7) |

| LSG    | RL LSG<br>M/C/O | Signal  | RL Signal<br>M/C/O | Option  |
|--------|-----------------|---|--------------------|---|
|        |                 | Number of valid serving sensors (A.1.3)   | M                  | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|        |                 | Size type: dynamic/fixed<br>Size #: Number of valid serving sensors (A.1.3)               |                    |   |
|        |                 | Sensor ID (A.1.4)   | M                  | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
|        |                 | Time stamp - <...> (A.1.5)  | M                  |   |
|        |                 | Message counter (A.1.6.2)   | O                  |   |
|        |                 | Interface cycle time (A.1.7)  | O                  |   |
|        |                 | Interface cycle time - variation (A.1.8)  | O                  |   |
|        |                 | Data qualifier (A.1.9)  | M                  | Default enumerators (B.1.7)   |
| Status | M               | Number of valid sensor operation modes (A.5.24)   | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor operation mode (A.5.25)   |
|        |                 | Size type: Dynamic/fixed<br>Size #: Number of valid sensor operation modes (A.5.24)       |                    |   |
|        |                 | Sensor operation mode (A.5.25)  | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|        |                 | Sensor defect recognised (A.5.26)   | M                  | Default enumerators (B.1.7)   |
|        |                 | Sensor defect reason (A.5.27)   | M                  | Default enumerators (B.1.7)   |
|        |                 | Supply voltage status (A.5.28)  | M                  | Default enumerators (B.1.7)   |
|        |                 | Sensor temperature status (A.5.29)  | M                  | Default enumerators (B.1.7)   |
|        |                 | Number of valid sensor input signal statuses (A.5.30)                                     | M                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor input signal type (A.5.31)                                      |
|        |                 | Size type: Dynamic/fixed<br>Size #: Number of valid sensor input signal statuses (A.5.30) |                    |   |
|        |                 | Sensor input signal type (A.5.31)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |


| LSG         | RL LSG<br>M/C/O   | Signal  | RL Signal<br>M/C/O   | Option  |
|-------------|---|---|--|---|
|             |   | Sensor input signal status (A.5.32)   | M  | Default enumerators (B.1.7)   |
|             |   | Sensor externally disturbed (A.5.33)  | O  | Default enumerators (B.1.7)   |
|             |   | Sensor transmit power reduced (A.5.34)  | O  | Default enumerators (B.1.7)   |
|             |   | Sensor heating status (A.5.35)  | O  | Default enumerators (B.1.7)   |
|             |   | Sensor cleaning status (A.5.36)   | O  | Default enumerators (B.1.7)   |
|             |   | Sensor time sync (A.5.37)   | O  | Default enumerators (B.1.7)   |
|             |   | Sensor time sync offset value (A.5.38)  | C (B.2.2)<br>* The signal "Sensor time sync" (A.5.37) has "STS_Offset" or a similar enumerator defined during the system design phase. |   |
| Calibration | O<br>Need of logical signal group (B.3.3)<br>1. SPI Header,<br>2. DLI Header,<br>3. FLI Header,<br>4. redundant on OLI Header | Number of valid sensor-calibratable components (A.5.39)                                     | M  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor-calibratable component (A.5.40) |
|             |   | Size type: Dynamic/fixed<br>Size #: Number of valid sensor-calibratable components (A.5.39) |  |   |
|             |   | Sensor-calibratable component (A.5.40)  | M  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)  |
|             |   | Sensor calibration status (A.5.41)  | M  | Default enumerators (B.1.7)   |
|             |   | Calibration process state (A.5.42)  | O  | Default enumerators (B.1.7)   |
|             |   | Sensor origin point – correction {x, y, z} (A.5.43)   | O  |   |
|             |   | Sensor origin point – correction {x, y, z} – error (A.5.44)                                 | O  | Implementation EM (B.4.1)   |

| LSG            | RL LSG<br>M/C/O  | Signal   | RL Signal<br>M/C/O | Option                    |
|----------------|--|--|--------------------|---------------------------|
|                |  | Sensor origin translation – correction limit {x <sub>begin</sub> , x <sub>end</sub> , y <sub>begin</sub> , y <sub>end</sub> , z <sub>begin</sub> , z <sub>end</sub> } (A.5.45)           | 0                  |                           |
|                |  | Sensor orientation – correction {yaw, pitch, roll} (A.5.46)  | 0                  |                           |
|                |  | Sensor orientation – correction {yaw, pitch, roll} – error (A.5.47)  | 0                  | Implementation EM (B.4.1) |
|                |  | Sensor pose angle – correction limit {yaw <sub>begin</sub> , yaw <sub>end</sub> , pitch <sub>begin</sub> , pitch <sub>end</sub> , roll <sub>begin</sub> , roll <sub>end</sub> } (A.5.48) | 0                  |                           |
| Sensor cluster | 0<br>Need of logical signal group (B.3.3)<br>1. SPI Header,<br>2. Redundant on OLI, FLI and DLI Header | Number of valid sensors (A.5.49)   | M                  |                           |
|                |  | Size type: Dynamic/fixed<br>Size #: Number of valid sensors (A.5.49)   |                    |                           |
|                |  | Sensor ID reference (A.5.50)   | M                  |                           |

#### 10.4.1 Sensor health information header

Table 54 defines the interface header in subclause “Sensor health information header” (10.4.1) and the changes due to the adaptation in comparison to the generic interface header in 10.2.1. The header of the SHII contains only one sensor information entity (see 10.4.2).

**Table 54 — Specific signal grouping: Sensor health information header**

| Signal  | RL signal | Option  |
|---|-----------|---|
| Interface version ID {major, minor, patch} (A.1.1)  | Mandatory | Profile: Uniqueness of interface versioning (6.4)   |
| Interface ID (A.1.2)  | Optional  | Profile: Uniqueness of interface versioning (6.4)<br>Default enumerators (B.1.7)  |
| Number of valid serving sensors (A.1.3)   | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Sensor ID (A.1.4) |
|  Sensor ID (A.1.4) | Mandatory | Profile: Uniqueness of interface versioning (6.4)<br>Alternative VRO (B.1.3)  |
| Time stamp – <...> (A.1.5)  | Mandatory |   |
| Message counter (A.1.6.2)   | Optional  |   |
| Interface cycle time (A.1.7)  | Optional  |   |
| Interface cycle time – variation (A.1.8)  | Optional  |   |
| Data qualifier (A.1.9)  | Mandatory | Default enumerators (B.1.7)   |

There is only one global entity in the SHII. Therefore, the sensor provides no list of SHII entities and a signal “Number of valid <...>” (A.1.12) is additionally not provided.

The following LSGs, which normally shall be provided by the sensor as well as the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor or sensor cluster (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SPI; LSG “Information: sensor pose” (see Table 51).

The following LSGs, which shall be provided by the sensor as well as the sensor cluster in this interface, may be added to another header (see B.3.3):

- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

#### 10.4.2 Sensor health information entity

Sensor health information of a sensor describes global sensor statuses and consists of several LSGs. This interface provides only one sensor health information entity for the entire sensor and not a list of entities.

- **Status:** the status describes the global statuses of the sensor.
- **Calibration:** the status describes the statuses and information of the sensor calibration.
- **Sensor cluster:** the sensor cluster defines the set of sensors which defines the sensor cluster.

### 11 Sensor input interface

#### 11.1 General

The sensor input interface (SII) provides information and parametrisation for a sensor or a sensor cluster. The sensor receives the SIIs via the in-vehicle communication. Each SII can be sent to the sensor or sensor cluster by different ECUs. Also, several ECUs can provide complementary parts of the interface (for example, one ECU provides weather information, and a second ECU provides road information of the CSH). During the system design phase, each received SIIs of a sensor or a sensor cluster and the ECUs that originate these interfaces shall be defined. The information source and the way how the signals are provided shall be defined during the system design phase to meet the safety and system design requirements (for example, that the fusion unit may provide certain SIIs).

#### 11.2 Generic sensor input interface

Table 55 defines the generic structure of a SII.

**Table 55 — Generic sensor input interface**

| Structure                             | Multiplicity | Option |
|---------------------------------------|--------------|--------|
| Generic sensor inputs header (11.2.1) | 1            |        |
| Generic sensor inputs entity (11.2.2) | 1            |        |

##### 11.2.1 Generic sensor inputs header

Table 56 defines the interface header in subclause “Generic sensor inputs header” (11.2.1) and the changes due to the adaptation in comparison to the generic interface header in 6.2.

**Table 56 — Specific signal grouping: Generic sensor detections header**

| Signal   | RL signal | Option   |
|--|-----------|--|
| Interface version ID {major, minor, patch} (A.1.1) | Mandatory | Profile: Uniqueness of interface versioning of SII (11.2.3)                                |
| Interface ID (A.1.2)                               | Optional  | Profile: Uniqueness of interface versioning of SII (11.2.3)<br>Default enumerators (B.1.7) |
| Sender ID (A.1.27)                                 | Mandatory | Profile: Uniqueness of interface versioning of SII (11.2.3)                                |
| Time stamp – <...> (A.1.5)                         | Mandatory |  |
| Message counter (A.1.6.2)                          | Optional  |  |
| Interface cycle time (A.1.7)                       | Optional  |  |
| Interface cycle time – variation (A.1.8)           | Optional  |  |
| Data qualifier (A.1.9)                             | Optional  | Default enumerators (B.1.7)  |

**11.2.2 Generic sensor inputs entity**

Each SII provides only one entity per interface. No generic LSG is defined for the interface entity types of all SIIs.

**11.2.3 Profile: Uniqueness of interface versioning of SII**

This profile adapts the profile of 6.4. A sensor or sensor cluster may receive this interface more than once (for example, different ECUs). To identify different senders of the SII a signal “Sender ID” (A.1.27) is used.

**11.3 Common sensor input interface**

The subclause “Common sensor input interface” (11.3) (CSII) provides the generic input of a sensor or sensor cluster.

Table 57 defines the logical structure of the common sensor input interface.

**Table 57 — Common sensor input interface**

| LSG                       | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O | Option   |
|---------------------------|-----------------|--|--------------------|--|
| Information:<br>interface | M               | Interface version ID {major, minor, patch} (A.1.1) | M                  | Profile: Uniqueness of interface versioning of SII (11.2.3)                                |
|                           |                 | Interface ID (A.1.2)                               | O                  | Profile: Uniqueness of interface versioning of SII (11.2.3)<br>Default enumerators (B.1.7) |
|                           |                 | Sender ID (A.1.27)                                 | M                  | Profile: Uniqueness of interface versioning of SII (11.2.3)                                |
|                           |                 | Time stamp – <...> (A.1.5)                         | M                  |  |
|                           |                 | Message counter (A.1.6.2)                          | O                  |  |

| LSG                                | RL LSG<br>M/C/O | Signal  | RL signal<br>M/C/O | Option   |
|------------------------------------|-----------------|---|--------------------|--|
|                                    |                 | Interface cycle time (A.1.7)  | 0                  |  |
|                                    |                 | Interface cycle time – variation (A.1.8)                                      | 0                  |  |
|                                    |                 | Data qualifier (A.1.9)  | 0                  | Default enumerators (B.1.7)  |
| Sensor operation                   | 0               | Sensor operation mode command (A.6.1)   | M                  | Default enumerators (B.1.7)  |
| Environmental information: weather | 0               | Weather condition (A.6.2)   | 0                  | Default enumerators (B.1.7)  |
|                                    |                 | Fog (A.6.3)   | 0                  | Default enumerators (B.1.7)  |
|                                    |                 | Precipitation (A.6.4)   | 0                  | Default enumerators (B.1.7)  |
|                                    |                 | Number of valid absorption models (A.6.5)                                     | 0                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Absorption wavelength (A.6.7) |
|                                    |                 | Size type: Dynamic/fixed<br>Size #: Number of valid absorption models (A.6.5) |                    |  |
|                                    |                 | Absorption coefficient (A.6.6)  | M                  |  |
|                                    |                 | Absorption wavelength (A.6.7)   | M                  | Alternative VRO (B.1.3)<br>Default enumerators (B.1.7)                               |
|                                    |                 | Ambient illumination (A.6.8)  | 0                  | Default enumerators (B.1.7)  |
|                                    |                 | Air temperature (A.6.9)   | M                  |  |
|                                    |                 | Air temperature – error (A.6.10)  | 0                  | Implementation EM (B.4.1)  |
|                                    |                 | Atmospheric pressure (A.6.11)   | 0                  |  |
|                                    |                 | Atmospheric pressure – error (A.6.12)   | 0                  | Implementation EM (B.4.1)  |
|                                    |                 | Relative humidity (A.6.13)  | 0                  |  |
|                                    |                 | Relative humidity – error (A.6.14)  | 0                  | Implementation EM (B.4.1)  |
| Environmental information: time    | 0               | Global time stamp (A.6.15)  | M                  |  |
| Environmental information: road    | 0               | Road type – input level (A.6.16)  | M                  | Default enumerators (B.1.7)  |
|                                    |                 | Road surface temperature (A.6.17)   | 0                  |  |
|                                    |                 | Road surface water film (A.6.18)  | 0                  |  |
|                                    |                 | Road surface freezing point (A.6.19)  | 0                  |  |
|                                    |                 | Road surface ice coverage (A.6.20)  | 0                  |  |
|                                    |                 | Road surface classification type – input level (A.6.21)                       | 0                  | Default enumerators (B.1.7)  |
|                                    |                 | Road surface roughness – input level (A.6.22)                                 | 0                  |  |

| LSG           | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O | Option                      |
|---------------|-----------------|--|--------------------|-----------------------------|
|               |                 | Road surface condition classification type – input level (A.6.23)        | 0                  | Default enumerators (B.1.7) |
|               |                 | Road surface texture – input level (A.6.24)                              | 0                  |                             |
| Vehicle state | 0               | Vehicle coordinate system type – input level (A.6.25)                    | M                  | Optimise IV (B.1.5)         |
|               |                 | Vehicle state reference point type (A.6.26)                              | 0                  |                             |
|               |                 | Velocity {x, y, z} – input level (A.6.27)                                | {M, M, O}          |                             |
|               |                 | Velocity {x, y, z} – error – input level (A.6.28)                        | 0                  | Implementation EM (B.4.1)   |
|               |                 | Acceleration {x, y, z} – input level (A.6.29)                            | 0                  |                             |
|               |                 | Acceleration {x, y, z} – error – input level (A.6.30)                    | 0                  | Implementation EM (B.4.1)   |
|               |                 | Jerk {x} (A.6.31)  | 0                  |                             |
|               |                 | Jerk {x} – error (A.6.32)  | 0                  | Implementation EM (B.4.1)   |
|               |                 | Rotation rate {yaw, pitch, roll} – input level (A.6.33)                  | {M, O, O}          |                             |
|               |                 | Rotation rate {yaw, pitch, roll} – error – input level (A.6.34)          | 0                  | Implementation EM (B.4.1)   |
|               |                 | Angular wheel position {FLP, FRP, RLP, RRP} (A.6.35)                     | 0                  |                             |
|               |                 | Angular wheel velocity {FLP, FRP, RLP, RRP} (A.6.36)                     | M                  |                             |
|               |                 | Angular wheel velocity {FLP, FRP, RLP, RRP} – error (A.6.37)             | 0                  | Implementation EM (B.4.1)   |
|               |                 | Slip angle {FLP, FRP, RLP, RRP} (A.6.38)                                 | 0                  |                             |
|               |                 | Active suspension {FLP, FRP, RLP, RRP} (A.6.39)                          | 0                  |                             |
|               |                 | Steering angle (A.6.40)  | 0                  |                             |
|               |                 | Steering angle – error (A.6.41)  | 0                  | Implementation EM (B.4.1)   |
|               |                 | Vehicle global position {latitude, longitude, altitude} (A.6.42)         | 0                  |                             |
|               |                 | Vehicle global position {latitude, longitude, altitude} – error (A.6.43) | 0                  | Implementation EM (B.4.1)   |
|               |                 | Vehicle global orientation {yaw, pitch, roll} (A.6.44)                   | 0                  |                             |
|               |                 | Wheel rotation direction {FLP, FRP, RLP, RRP} (A.6.45)                   | 0                  | Default enumerators (B.1.7) |



| LSG                   | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O | Option   |
|-----------------------|-----------------|--|--------------------|--|
|                       |                 | Wheel tick count {FLP, FRP, RLP, RRP} (A.6.46)   | O                  | Redundancy (B.1.2)<br>Signal: Angular wheel position {FLP, FRP, RLP, RRP} (A.6.35) or Angular wheel velocity {FLP, FRP, RLP, RRP} (A.6.36)                           |
|                       |                 | Gear position (A.6.47)   | O                  | Default enumerators (B.1.7)  |
|                       |                 | Brake activation level (A.6.48)  | O                  | Default enumerators (B.1.7)  |
|                       |                 | Vehicle motion state (A.6.49)  | O                  | Default enumerators (B.1.7)  |
| Region information    | O               | Current region (A.6.50)  | M                  | Default enumerators (B.1.7)  |
|                       |                 | Current traffic flow type (A.6.51)   | M                  | Default enumerators (B.1.7)  |
| Vehicle configuration | O               | Trailer type (A.6.52)  | O                  | Default enumerators (B.1.7)  |
|                       |                 | Park brake state (A.6.53)  | O                  | Default enumerators (B.1.7)  |
|                       |                 | Vehicle body position {z} (A.6.54)   | M                  | Optimise IV (B.1.5)  |
|                       |                 | Vehicle body orientation {pitch, roll} (A.6.55)  | O                  |  |
| Sensor pose           | O               | Sensor origin point {x, y, z} – input level (A.6.56)                                     | M                  | Optimise IV (B.1.5)  |
|                       |                 | Sensor orientation {yaw, pitch, roll} – input level (A.6.57)                             | M                  | Optimise IV (B.1.5)  |
| PMOI operation        | O               | Reporting interface PMOI – command (A.6.58.1)  | M                  | Default enumerators (B.1.7)  |
| RDOI operation        | O               | Reporting interface RDOI – command (A.6.58.2)  | M                  | Default enumerators (B.1.7)  |
| SOI operation         | O               | Reporting interface SOI – command (A.6.58.3)   | M                  | Default enumerators (B.1.7)  |
| FSAOI operation       | O               | Reporting interface FSAOI – command (A.6.58.4)   | M                  | Default enumerators (B.1.7)  |
| FLI operation         | O               | Reporting interface FLI – command (A.6.58.5)   | M                  | Default enumerators (B.1.7)  |
| DLI operation         | O               | Reporting interface DLI – command (A.6.58.6)   | M                  | Default enumerators (B.1.7)  |
| SPI operation         | O               | Reporting interface SPI – command (A.6.58.7)   | M                  | Default enumerators (B.1.7)  |
|                       |                 | Number of valid region-of-interest segments (A.6.59)                                     | O                  | Optimise LL (B.1.1)<br>Alternative UTL (B.1.6)<br>Key: Segment azimuth – input level {begin, end} (A.6.60) and Segment elevation – input level {begin, end} (A.6.61) |
|                       |                 | Size type: Dynamic/fixed<br>Size #: Number of valid region-of-interest segments (A.6.59) |                    |  |

| LSG            | RL LSG<br>M/C/O | Signal   | RL signal<br>M/C/O | Option                      |
|----------------|-----------------|--|--------------------|-----------------------------|
|                |                 | Segment azimuth – input level<br>{begin, end} (A.6.60)   | O                  | Alternative VRO (B.1.3)     |
|                |                 | Segment elevation – input level<br>{begin, end} (A.6.61) | O                  | Alternative VRO (B.1.3)     |
|                |                 | Resolution type {azimuth,<br>elevation} (A.6.62)         | M                  | Default enumerators (B.1.7) |
| SHII operation | O               | Reporting interface SHII – command<br>(A.6.58.8)         | M                  | Default enumerators (B.1.7) |

### 11.3.1 Common sensor input header

Table 58 defines the interface header in subclause “Common sensor input header” (11.3.1) and the changes due to the adaptation in comparison to the generic sensor inputs header in 11.2.1.

**Table 58 — Specific signal grouping: Sensor input header**

| Signal   | RL signal | Option   |
|--|-----------|--|
| Interface version ID {major, minor, patch} (A.1.1) | Mandatory | Profile: Uniqueness of interface versioning of SII (11.2.3)                                |
| Interface ID (A.1.2)                               | Optional  | Profile: Uniqueness of interface versioning of SII (11.2.3)<br>Default enumerators (B.1.7) |
| Sender ID (A.1.27)                                 | Mandatory | Profile: Uniqueness of interface versioning of SII (11.2.3)                                |
| Time stamp – <...> (A.1.5)                         | Mandatory |  |
| Message counter (A.1.6.2)                          | Optional  |  |
| Interface cycle time (A.1.7)                       | Optional  |  |
| Interface cycle time – variation (A.1.8)           | Optional  |  |
| Data qualifier (A.1.9)                             | Optional  | Default enumerators (B.1.7)  |

The signal “Time stamp – <...>” (A.1.5) shall be defined during the system design phase.

### 11.3.2 Common sensor input entity

The common sensor input for a sensor describes global ego-vehicle information and requests sensor parameterisation for the interfaces and consists of several LSGs. This interface provides only one common sensor input entity for the entire sensor and not a list of entities.

- **Sensor operation:** the sensor operation defines the global operation mode for the sensor.
- **Environmental information:** the environmental information weather, time and road describe the dynamic environmental information for the ego-vehicle.
- **Vehicle state:** the vehicle state describes the dynamic states of the ego-vehicle.
- **Region information:** the region information describes the geographical region in which the ego-vehicle is currently driving.

- **Vehicle configuration:** the vehicle configuration describes the ego-vehicle and the attached trailer to the ego-vehicle.
- **Sensor pose:** the sensor pose describes the sensor coordinate system in the ego-vehicle coordinate system.
- **<Interface> operation:** the <interface> operation defines operation mode for each of the sensor's interfaces and provides additional parametrisations (for example, the LSG "SPI operation" defines the resolution type for field-of-view segments).

STANDARDSISO.COM : Click to view the full PDF of ISO 23150:2023

## Annex A (normative)

### Interface signals

#### A.1 Header signals

Interface header specific signals are defined in this subclause (see Table A.1-Table A.67). If different definitions are required for different interfaces, the interface-specific signal is defined in a subclause, for example, the signal “Time stamp – <...>” (A.1.5) and the specific signal “Time stamp – prediction” (A.1.5.1).

##### A.1.1 Interface version ID {major, minor, patch}

**Table A.1 — Signal: Interface version ID {major, minor, patch}**

|                    |   |             |           |
|--------------------|---|-------------|-----------|
| <b>Name</b>        | Interface version ID {major, minor, patch}  |             |           |
| <b>Description</b> | <p>The signal “Interface version ID {major, minor, patch}” (A.1.1) provides the version information of the interface.</p> <p>Requirement:<br/>The signal “Interface version ID {major, minor, patch}” (A.1.1) is assigned in the system design phase.</p> |             |           |
| <b>Value type</b>  | 3D vector value   | <b>Unit</b> | (1, 1, 1) |

##### A.1.2 Interface ID

**Table A.2 — Signal: Interface ID**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Interface ID   |             |   |
| <b>Description</b> | <p>The signal “Interface ID” (A.1.2) is used to uniquely identify the originated interface type of the message.</p> <p>Additional information:<br/>The enumerators may differentiate the sensor’s sensor technology.<br/>The signal “Interface ID” (A.1.2) is necessary, for example, for a service-oriented architecture.</p> |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.3 — Enumeration: Interface ID - Example enumerators**

| <b>Name</b>                  | <b>Description</b>   | <b>RL enumerator</b> |
|------------------------------|--|----------------------|
| IID_PotentiallyMovingObjects | The message was sent by PMOI.  | “exemplary”          |
| IID_RoadObjects              | The message was sent by RDOI.  | “exemplary”          |
| IID_StaticObjects            | The message was sent by SOI.   | “exemplary”          |
| IID_FreeSpaceAreaObjects     | The message was sent by FSAOI.   | “exemplary”          |
| IID_CameraFeatures           | The message was sent by the sensor technology specific FLI. The sensor or sensor cluster uses camera sensing elements. | “exemplary”          |

|                             |  |             |
|-----------------------------|--|-------------|
| IID_UltrasonicFeatures      | The message was sent by the sensor technology specific FLI. The sensor or sensor cluster uses ultrasonic sensing elements. | "exemplary" |
| IID_RadarDetections         | The message was sent by the sensor technology specific DLI. The sensor or sensor cluster uses radar sensing elements.      | "exemplary" |
| IID_LidarDetections         | The message was sent by the sensor technology specific DLI. The sensor or sensor cluster uses lidar sensing elements.      | "exemplary" |
| IID_CameraDetections        | The message was sent by the sensor technology specific DLI. The sensor or sensor cluster uses camera sensing elements.     | "exemplary" |
| IID_UltrasonicDetections    | The message was sent by the sensor technology specific DLI. The sensor or sensor cluster uses ultrasonic sensing elements. | "exemplary" |
| IID_SensorPerformance       | The message was sent by SPI.   | "exemplary" |
| IID_SensorHealthInformation | The message was sent by SHII.  | "exemplary" |
| IID_CommonSensorInput       | The message is sent to the sensor or sensor cluster as input for the sensor.   | "exemplary" |

### A.1.3 Number of valid serving sensors

**Table A.4 — Signal: Number of valid serving sensors**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid serving sensors  |             |   |
| <b>Description</b> | The signal "Number of valid serving sensors" (A.1.3) provides the number of valid sensors serving the interface. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

### A.1.4 Sensor ID

**Table A.5 — Signal: Sensor ID**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Sensor ID  |             |   |
| <b>Description</b> | <p>The signal "Sensor ID" (A.1.4) which uniquely identifies the sensor and link the data of different sensors' interfaces.</p> <p>Additional information:</p> <p>It is required to associate the logical interfaces of one sensor or a sensor cluster with its sensors.</p> <p>Each signal "Sensor ID" (A.1.4) value shall serve independently at least one interface.</p> <p>The signal "Sensor ID" (A.1.4) is necessary, for example, for a service-oriented architecture.</p> <p>Requirement:</p> <p>The signal's "Sensor ID" (A.1.4) value is assigned during the system design phase.</p> |             |   |
| <b>Value type</b>  | integer value  | <b>Unit</b> | 1 |

### A.1.5 Time stamp – <...>

The time stamp is defined depending on the level of the interface. It is either the most consistent time of the measurement or the predicted time at which the message is consistent. An additional time stamp difference specifies an individual measurement point with a time offset of each entity relative to the time stamp of the message. The time stamps can also be used to link the data of different interfaces of a sensor cluster.

#### A.1.5.1 Time stamp – prediction

**Table A.6 — Signal: Time stamp – prediction**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Time stamp – prediction   |             |   |
| <b>Description</b> | <p>The signal “Time stamp – prediction” (A.1.5.1) provides the time stamp at which the entities of the interface are consistent (not the time at which it was processed or at which it is transmitted) in the vehicle-global synchronised time. Clock synchronisation shall be ensured, depending on bus technology.</p> <p>Additional information:</p> <p>The time stamp that was used for the prediction or tracking. Specifically, the time stamp can be the same as or differ from the base time stamp(s) of the, for example, underlying detection or feature’s entity list(s).</p> <p>The signal is used for tracked or predicted quantity values.</p> <p>Time stamps are consistent for all vehicle electronic control units (ECUs).</p> |             |   |
| <b>Value type</b>  | real value  | <b>Unit</b> | s |

#### A.1.5.2 Time stamp – measurement

**Table A.7 — Signal: Time stamp – measurement**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Time stamp – measurement   |             |   |
| <b>Description</b> | <p>The signal “Time stamp – measurement” (A.1.5.2) provides the time stamp at which the measurement was taken (not the time at which it was processed or at which it is transmitted) in the vehicle-global synchronised time. Clock synchronisation shall be ensured, depending on bus technology.</p> <p>Additional information:</p> <p>The signal’s “Time stamp – measurement” (A.1.5.2) value shall be selected so that it is as consistent as possible with the measured entities and the associated time error is minimum.</p> <p>Each entity of a detection list may have, for example, the signal “Time stamp difference – detection level” (A.4.5) for its measurement (for example, for integrating sensor technologies).</p> <p>In case of a continuous or a subdivided measurement cycle, the signal “Time stamp difference – detection level” (A.4.5) shall be used to differentiate the real measured point in time of the entity acquisition.</p> <p>Time stamps are consistent for all vehicle electronic control units (ECUs).</p> |             |   |
| <b>Value type</b>  | real value   | <b>Unit</b> | s |

#### A.1.6 <...> counter

A continuous value that defines the order of the logical interface’s messages. The logical interface messages may be sent cyclically or non-cyclically, for example, event triggered.

## A.1.6.1 Cycle counter

Table A.8 — Signal: Cycle counter

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Cycle counter  |             |   |
| <b>Description</b> | <p>The signal “Cycle counter” (A.1.6.1) provides the continuous up counter to identify the message cycle of the interface. It counts up per every cycle of a complete prediction, tracking or a single measurement.</p> <p>Additional information:</p> <p>This information can be determined from signal’s “Time stamp – &lt;...&gt;” (A.1.5) value and the sensor’s cycle period [see the signal “Interface cycle time” (A.1.7) and “Interface cycle time – variation” (A.1.8)].</p> <p>Each sensor as well as each sensor cluster uses a local sensor cycle counter per interface.</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

## A.1.6.2 Message counter

Table A.9 — Signal: Message counter

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Message counter  |             |   |
| <b>Description</b> | <p>The signal “Message counter” (A.1.6.2) provides the continuous up counter to identify the message sequence of the interface. It counts up per every sent message.</p> <p>Additional information:</p> <p>Interfaces with a signal “Message counter” (A.1.6.2) may send messages non-cyclical, for example, event triggered.</p> <p>Each sensor as well as each sensor cluster uses a local sensor message counter per interface.</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

## A.1.7 Interface cycle time

Table A.10 — Signal: Interface cycle time

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Interface cycle time  |             |   |
| <b>Description</b> | <p>The signal “Interface cycle time” (A.1.7) provides the representative cycle time of the interface’s messages.</p> <p>Normally, each interface sends the complete interface message cyclically. The period between two messages can vary. Each interface header defines the interface’s cycle time and the variation of the cycle time [see signal “Interface cycle time – variation” (A.1.8)].</p> <p>Additional information:</p> <p>If the signal “Interface cycle time – variation” (A.1.8) is 100 %, the message is sent in the case one or several signals have changed their value.</p> |             |   |
| <b>Value type</b>  | [0...] real value   | <b>Unit</b> | s |

## A.1.8 Interface cycle time – variation

Table A.11 — Signal: Interface cycle time – variation

|             |                                  |  |  |
|-------------|----------------------------------|--|--|
| <b>Name</b> | Interface cycle time – variation |  |  |
|-------------|----------------------------------|--|--|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | <p>The signal “Interface cycle time – variation” (A.1.8) provides the possible variation of the expected representative cycle time [see signal “Interface cycle time” (A.1.7)] of the interface’s messages.</p> <p>Additional information:</p> <p>If the signal “Interface cycle time – variation” (A.1.8) is 100 %, the message is sent in the case one or several signals have changed their value, for example, the possible, allowed cycle time for the next cycle: <math>0 \text{ s} \leq \text{next cycle time} \leq \text{infinity s}</math>. This may indicate that the next cycle is triggered by an event.</p> <p>EXAMPLE The duration between two sequentially sent messages of an interface is in the range <math>[\text{interface cycle time} \times (1 - \text{interface cycle time variation} / 100 \%), \text{interface cycle time} / (1 - \text{interface cycle time variation} / 100 \%)]</math>.</p> |             |   |
| <b>Value type</b>  | [0...] real value   | <b>Unit</b> | % |

### A.1.9 Data qualifier

**Table A.12 — Signal: Data qualifier**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Data qualifier   |             |   |
| <b>Description</b> | The signal “Data qualifier” (A.1.9) expresses whether the content of the entity list(s) of the interface can be used or not. It can change dynamically over time and inform about temporal restrictions. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.13 — Enumeration: Data qualifier – Example enumerators**

| Name                              | Description  | RL enumerator |
|-----------------------------------|--|---------------|
| DQ_Normal                         | The entities and information of the interface can be used without restrictions.  | “exemplary”   |
| DQ_NotAvailable                   | The entities and information of the interface are not available.   | “exemplary”   |
| DQ_ReduceInCoverage               | The entities are not complete due to a restricted coverage (for example, restricted view).                               | “exemplary”   |
| DQ_ReduceInPerformance            | The entities are not complete due to a restricted performance.   | “exemplary”   |
| DQ_ReduceInCoverageAndPerformance | The entities are not complete due to a restricted coverage and restricted performance.                                   | “exemplary”   |
| DQ_TestMode                       | The sensor or the sensor cluster is running in test mode and outputs differ from normal operation mode.                  | “exemplary”   |
| DQ_Invalid                        | The measurement, tracking or prediction cycle was invalid, no predicted, tracked or measured entity will be transferred. | “exemplary”   |

### A.1.10 Recognised <...> – capability

The performance of the sensor as well as the sensor cluster determines how many entities the sensor cluster’s interface can recognise (except SHII), for example, at full load. For example, a value of 0 informs that the sensor may not recognise any object/feature/detection at the moment.



**A.1.10.1 Recognised potentially moving objects – capability****Table A.14 — Signal: Recognised potentially moving objects – capability**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Recognised potentially moving objects – capability  |             |   |
| <b>Description</b> | The signal “Recognised potentially moving objects – capability” (A.1.10.1) provides the performance of the sensor cluster’s interface. It determines how many entities the sensor cluster is guaranteed to recognise, for example, at full load with respect to the current scenario. |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.1.10.2 Recognised road markings – capability****Table A.15 — Signal: Recognised road markings – capability**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Recognised road markings – capability  |             |   |
| <b>Description</b> | The signal “Recognised road markings – capability” (A.1.10.2) provides the performance of the sensor cluster’s interface. It determines how many entities the sensor cluster is guaranteed to recognise, for example, at full load with respect to the current scenario. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.1.10.3 Recognised road boundaries – capability****Table A.16 — Signal: Recognised road boundaries – capability**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Recognised road boundaries – capability  |             |   |
| <b>Description</b> | The signal “Recognised road boundaries – capability” (A.1.10.3) provides the performance of the sensor cluster’s interface. It determines how many entities the sensor cluster is guaranteed to recognise, for example, at full load with respect to the current scenario. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.1.10.4 Recognised general landmarks – capability****Table A.17 — Signal: Recognised general landmarks – capability**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Recognised general landmarks – capability  |             |   |
| <b>Description</b> | The signal “Recognised general landmarks – capability” (A.1.10.4) provides the performance of the sensor cluster’s interface. It determines how many entities the sensor cluster is guaranteed to recognise, for example, at full load with respect to the current scenario. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.1.10.5 Recognised traffic signs – capability****Table A.18 — Signal: Recognised traffic signs – capability**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Recognised traffic signs – capability  |             |   |
| <b>Description</b> | The signal “Recognised traffic signs – capability” (A.1.10.5) provides the performance of the sensor cluster’s interface. It determines how many entities the sensor cluster is guaranteed to recognise, for example, at full load with respect to the current scenario. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

## A.1.10.6 Recognised traffic sign boards – capability

Table A.19 — Signal: Recognised traffic sign boards – capability

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Recognised traffic sign boards – capability  |             |   |
| <b>Description</b> | The signal “Recognised traffic sign boards – capability” (A.1.10.6) provides the performance of the sensor cluster’s interface. It determines how many entities the sensor cluster is guaranteed to recognise, for example, at full load with respect to the current scenario. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

## A.1.10.7 Recognised traffic lights – capability

Table A.20 — Signal: Recognised traffic lights – capability

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Recognised traffic lights – capability  |             |   |
| <b>Description</b> | The signal “Recognised traffic lights – capability” (A.1.10.7) provides the performance of the sensor cluster’s interface. It determines how many entities the sensor cluster is guaranteed to recognise, for example, at full load with respect to the current scenario. |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

## A.1.10.8 Recognised free space areas – capability

Table A.21 — Signal: Recognised free space areas – capability

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Recognised free space areas – capability  |             |   |
| <b>Description</b> | The signal “Recognised free space areas – capability” (A.1.10.8) provides the performance of the sensor cluster’s interface. It determines how many entities the sensor cluster is guaranteed to recognise, for example, at full load with respect to the current scenario. |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

## A.1.10.9 Recognised features – capability

Table A.22 — Signal: Recognised features – capability

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Recognised features – capability  |             |   |
| <b>Description</b> | The signal “Recognised features – capability” (A.1.10.9) provides the performance of the sensor cluster’s interface. It determines how many entities the sensor cluster is guaranteed to recognise, for example, at full load with respect to the current scenario. |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

## A.1.10.10 Recognised detections – capability

Table A.23 — Signal: Recognised detections – capability

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Recognised detections – capability   |             |   |
| <b>Description</b> | The signal “Recognised detections – capability” (A.1.10.10) provides the performance of the sensor cluster’s interface. It determines how many entities the sensor cluster is guaranteed to recognise, for example, at full load with respect to the current scenario. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.1.10.11 Recognised shapes – capability****Table A.24 — Signal: Recognised shapes – capability**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Recognised shapes – capability   |             |   |
| <b>Description</b> | The signal “Recognised shapes – capability” (A.1.10.11) provides the performance of the sensor cluster’s interface. It determines how many entities the sensor cluster is guaranteed to recognise, for example, at full load with respect to the current scenario. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.1.11 Recognised <...> – status**

The performance of the sensor cluster’s interface determines a state of the recognition process under, for example, full load (except SHII).

All adaptations of the signal use the same list of example enumerators.

**Table A.25 — Enumeration: Recognised <...> – status – Example enumerators**

| <b>Name</b>       | <b>Description</b>   | <b>RL enumerator</b> |
|-------------------|--|----------------------|
| R<...>S_Normal    | The performance of the sensor’s or the sensor cluster’s interface is enough to process all recognised entities.              | “exemplary”          |
| R<...>S_PreLimits | The performance of the sensor’s or the sensor cluster’s interface is close to the limits to process all recognised entities. | “exemplary”          |
| R<...>S_Limits    | The performance of the sensor’s or the sensor cluster’s interface is not enough to process all recognised entities.          | “exemplary”          |

**A.1.11.1 Recognised potentially moving objects – status****Table A.26 — Signal: Recognised potentially moving objects – status**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Recognised potentially moving objects – status   |             |   |
| <b>Description</b> | The signal “Recognised potentially moving objects – status” (A.1.11.1) determines whether the sensor cluster’s interface can process all recognised entities or whether it works at the sensor cluster’s performance limits. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**A.1.11.2 Recognised road markings – status****Table A.27 — Signal: Recognised road markings – status**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Recognised road markings – status   |             |   |
| <b>Description</b> | The signal “Recognised road markings – status” (A.1.11.2) determines whether the sensor cluster’s interface can process all recognised entities or whether it works at the sensor cluster’s performance limits. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**A.1.11.3 Recognised road boundaries – status****Table A.28 — Signal: Recognised road boundaries – status**

|             |                                     |  |  |
|-------------|-------------------------------------|--|--|
| <b>Name</b> | Recognised road boundaries – status |  |  |
|-------------|-------------------------------------|--|--|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | The signal “Recognised road boundaries – status” (A.1.11.3) determines whether the sensor cluster’s interface can process all recognised entities or whether it works at the sensor cluster’s performance limits. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

#### A.1.11.4 Recognised general landmarks – status

**Table A.29 — Signal: Recognised general landmarks – status**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Recognised general landmarks – status   |             |   |
| <b>Description</b> | The signal “Recognised general landmarks – status” (A.1.11.4) determines whether the sensor cluster’s interface can process all recognised entities or whether it works at the sensor cluster’s performance limits. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

#### A.1.11.5 Recognised traffic signs – status

**Table A.30 — Signal: Recognised traffic signs – status**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Recognised traffic signs – status   |             |   |
| <b>Description</b> | The signal “Recognised traffic signs – status” (A.1.11.5) determines whether the sensor cluster’s interface can process all recognised entities or whether it works at the sensor cluster’s performance limits. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

#### A.1.11.6 Recognised traffic sign boards – status

**Table A.31 — Signal: Recognised traffic sign boards – status**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Recognised traffic sign boards – status   |             |   |
| <b>Description</b> | The signal “Recognised traffic sign boards – status” (A.1.11.6) determines whether the sensor cluster’s interface can process all recognised entities or whether it works at the sensor cluster’s performance limits. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

#### A.1.11.7 Recognised traffic lights – status

**Table A.32 — Signal: Recognised traffic lights – status**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Recognised traffic lights – status   |             |   |
| <b>Description</b> | The signal “Recognised traffic lights – status” (A.1.11.7) determines whether the sensor cluster’s interface can process all recognised entities or whether it works at the sensor cluster’s performance limits. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

#### A.1.11.8 Recognised free space areas – status

**Table A.33 — Signal: Recognised free space areas – status**

|                    |  |  |  |
|--------------------|--|--|--|
| <b>Name</b>        | Recognised free space areas – status   |  |  |
| <b>Description</b> | The signal “Recognised free space areas – status” (A.1.11.8) determines whether the sensor cluster’s interface can process all recognised entities or whether it works at the sensor cluster’s performance limits. |  |  |

|                   |             |             |   |
|-------------------|-------------|-------------|---|
| <b>Value type</b> | enumeration | <b>Unit</b> | 1 |
|-------------------|-------------|-------------|---|

#### A.1.11.9 Recognised features – status

**Table A.34 — Signal: Recognised features – status**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Recognised features – status   |             |   |
| <b>Description</b> | The signal “Recognised features – status” (A.1.11.9) determines whether the sensor cluster’s interface can process all recognised entities or whether it works at the sensor cluster’s performance limits. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

#### A.1.11.10 Recognised detections – status

**Table A.35 — Signal: Recognised detections – status**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Recognised detections – status  |             |   |
| <b>Description</b> | The signal “Recognised detections – status” (A.1.11.10) determines whether the sensor cluster’s interface can process all recognised entities or whether it works at the sensor cluster’s performance limits. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

#### A.1.11.11 Recognised shapes – status

**Table A.36 — Signal: Recognised shapes – status**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Recognised shapes – status  |             |   |
| <b>Description</b> | The signal “Recognised shapes – status” (A.1.11.11) determines whether the sensor cluster’s interface can process all recognised entities or whether it works at the sensor cluster’s performance limits. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

#### A.1.12 Number of valid <...>

Each interface (except SHII) sends one or more lists of entities with defined entity types. For each list, the number of valid transferred entities is known.

##### A.1.12.1 Number of valid potentially moving objects

**Table A.37 — Signal: Number of valid potentially moving objects**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid potentially moving objects   |             |   |
| <b>Description</b> | <p>The signal’s “Number of valid potentially moving objects” (A.1.12.1) value defines the number of valid entities in the potentially moving object list of the PMOI.</p> <p>Additional information:</p> <p>The entries in the list may or may not be sorted.</p> <p>All those entities of the list which can be used by the fusion unit are regarded as valid.</p> <p>The list includes a complete and prioritised set of valid entities for the signal’s “Time stamp – &lt;...&gt;” (A.1.5) value and not only a differential list between two sequential time stamps.</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

## A.1.12.2 Number of valid road markings

Table A.38 — Signal: Number of valid road markings

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid road markings   |             |   |
| <b>Description</b> | <p>The signal's "Number of valid road markings" (A.1.12.2) value defines the number of valid entities in the road marking list of the RDOI.</p> <p>Additional information:</p> <p>The entries in the list may or may not be sorted.</p> <p>All those entities of the list which can be used by the fusion unit are regarded as valid.</p> <p>The list includes a complete and prioritised set of valid entities for the signal's "Time stamp – &lt;...&gt;" (A.1.5) value and not only a differential list between two sequential time stamps.</p> <p>Each road marking has polynomial or polyline, for example, containing <math>\geq 2</math> vertices.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

## A.1.12.3 Number of valid road boundaries

Table A.39 — Signal: Number of valid road boundaries

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid road boundaries   |             |   |
| <b>Description</b> | <p>The signal's "Number of valid road boundaries" (A.1.12.3) value defines the number of valid entities in the road boundary list of the RDOI.</p> <p>Additional information:</p> <p>The entries in the list may or may not be sorted.</p> <p>All those entities of the list which can be used by the fusion unit are regarded as valid.</p> <p>The list includes a complete and prioritised set of valid entities for the signal's "Time stamp – &lt;...&gt;" (A.1.5) value and not only a differential list between two sequential time stamps.</p> <p>Each road boundary has polynomial or polyline, for example, containing <math>\geq 2</math> vertices.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

## A.1.12.4 Number of valid general landmarks

Table A.40 — Signal: Number of valid general landmarks

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid general landmarks   |             |   |
| <b>Description</b> | <p>The signal's "Number of valid general landmarks" (A.1.12.4) value defines the number of valid entities in the general landmark list of the SOI.</p> <p>Additional information:</p> <p>The entries in the list may or may not be sorted.</p> <p>All those entities of the list which can be used by the fusion unit are regarded as valid.</p> <p>The list includes a complete and prioritised set of valid entities for the signal's "Time stamp – &lt;...&gt;" (A.1.5) value and not only a differential list between two sequential time stamps.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.1.12.5 Number of valid traffic signs****Table A.41 — Signal: Number of valid traffic signs**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid traffic signs   |             |   |
| <b>Description</b> | <p>The signal's "Number of valid traffic signs" (A.1.12.5) value defines the number of valid entities in the traffic sign list of the SOI.</p> <p>Additional information:</p> <p>The entries in the list may or may not be sorted.</p> <p>All those entities of the list which can be used by the fusion unit are regarded as valid.</p> <p>The list includes a complete and prioritised set of valid entities for the signal's "Time stamp – &lt;...&gt;" (A.1.5) value and not only a differential list between two sequential time stamps.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.1.12.6 Number of valid traffic sign boards****Table A.42 — Signal: Number of valid traffic sign boards**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid traffic sign boards   |             |   |
| <b>Description</b> | <p>The signal's "Number of valid traffic sign boards" (A.1.12.6) value defines the number of valid entities in the traffic sign board list of the SOI.</p> <p>Additional information:</p> <p>The entries in the list may or may not be sorted.</p> <p>All those entities of the list which can be used by the fusion unit are regarded as valid.</p> <p>The list includes a complete and prioritised set of valid entities for the signal's "Time stamp – &lt;...&gt;" (A.1.5) value and not only a differential list between two sequential time stamps.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.1.12.7 Number of valid traffic lights****Table A.43 — Signal: Number of valid traffic lights**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid traffic lights  |             |   |
| <b>Description</b> | <p>The signal's "Number of valid traffic lights" (A.1.12.7) value defines the number of valid entities in the traffic light list of the SOI.</p> <p>Additional information:</p> <p>The entries in the list may or may not be sorted.</p> <p>All those entities of the list which can be used by the fusion unit are regarded as valid.</p> <p>The list includes a complete and prioritised set of valid entities for the signal's "Time stamp – &lt;...&gt;" (A.1.5) value and not only a differential list between two sequential time stamps.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.1.12.8 Number of valid free space areas****Table A.44 — Signal: Number of valid free space areas**

|             |                                  |  |  |
|-------------|----------------------------------|--|--|
| <b>Name</b> | Number of valid free space areas |  |  |
|-------------|----------------------------------|--|--|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | <p>The signal's "Number of valid free space areas" (A.1.12.8) value defines the number of valid entities in the free space area list of the FSAOI.</p> <p>Additional information:</p> <p>The entries in the list may or may not be sorted.</p> <p>All those entities of the list which can be used by the fusion unit are regarded as valid.</p> <p>The list includes a complete and prioritised set of valid entities for the signal's "Time stamp – &lt;...&gt;" (A.1.5) value and not only a differential list between two sequential time stamps.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

#### A.1.12.9 Number of valid features

**Table A.45 — Signal: Number of valid features**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid features  |             |   |
| <b>Description</b> | <p>The signal's "Number of valid features" (A.1.12.9) value defines the number of valid entities in the feature list of an FLI.</p> <p>One measurement can generate not only unique features, if there are, for example, ambiguities in camera shape or colour classification.</p> <p>Additional information:</p> <p>The entries in the list may or may not be sorted.</p> <p>All those entities of the list which can be used by the fusion unit are regarded as valid.</p> <p>The list includes a complete and prioritised set of valid entities for the signal's "Time stamp – &lt;...&gt;" (A.1.5) value and not only a differential list between two sequential time stamps.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

#### A.1.12.10 Number of valid detections

**Table A.46 — Signal: Number of valid detections**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid detections  |             |   |
| <b>Description</b> | <p>The signal's "Number of valid detections" (A.1.12.10) value defines the number of valid entities in the detection list of a DLI.</p> <p>One measurement can generate more than one similar detection, if there are ambiguities.</p> <p>Additional information:</p> <p>The entries in the list may or may not be sorted.</p> <p>All those entities of the list which can be used by the fusion unit are regarded as valid.</p> <p>The list includes a complete and prioritised set of valid entities for the signal's "Time stamp – &lt;...&gt;" (A.1.5) value and not only a differential list between two sequential time stamps.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

#### A.1.12.11 Number of valid shapes

**Table A.47 — Signal: Number of valid shapes**

|             |                        |  |  |
|-------------|------------------------|--|--|
| <b>Name</b> | Number of valid shapes |  |  |
|-------------|------------------------|--|--|



|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | <p>The signal's "Number of valid shapes" (A.1.12.11) value defines the number of valid entities in the detection list of the CDI or CFI.</p> <p>Additional information:</p> <p>The entries in the list may or may not be sorted.</p> <p>All those entities of the list which can be used by the fusion unit are regarded as valid.</p> <p>The list includes a complete and prioritised set of valid entities for the signal's "Time stamp - &lt;...&gt;" (A.1.5) value and not only a differential list between two sequential time stamps.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

#### A.1.12.12 Number of valid field-of-view segments

**Table A.48 — Signal: Number of valid field-of-view segments**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid field-of-view segments  |             |   |
| <b>Description</b> | <p>The signal's "Number of valid field-of-view segments" (A.1.12.12) value defines the number of FOVs sensor- or sensor cluster-segments of the SPI.</p> <p>Additional information:</p> <p>The entries in the list may or may not be sorted.</p> <p>All those entities of the list which can be used by the fusion unit are regarded as valid.</p> <p>The list includes a complete and prioritised set of valid entities for the signal's "Time stamp - &lt;...&gt;" (A.1.5) value and not only a differential list between two sequential time stamps.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

#### A.1.13 Motion type

**Table A.49 — Signal: Motion type**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Motion type   |             |   |
| <b>Description</b> | The signal "Motion type" (A.1.13) provides the applied motion type of the interface's motion signals. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.50 — Enumeration: Motion type – Example enumerators**

| <b>Name</b> | <b>Description</b>  | <b>RL enumerator</b> |
|-------------|---|----------------------|
| MT_Relative | The interface sends relative motion values of the object described in ego-vehicle coordinate system {x, y, z}.                              | "exemplary"          |
| MT_Absolute | The interface sends absolute motion values of the object in world coordinate reference system, using the ego-vehicle axis system {x, y, z}. | "exemplary"          |

#### A.1.14 Colour model type

**Table A.51 — Signal: Colour model type**

|             |                   |
|-------------|-------------------|
| <b>Name</b> | Colour model type |
|-------------|-------------------|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | <p>The signal “Colour model type” (A.1.14) provides the applied colour model for the colour signals of the interface.</p> <p>Additional information:</p> <p>The signal “Colour model type” (A.1.14) defines also the number of elements for the signal “Colour values” (see A.2.100, A.3.12, A.4.29).</p> |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.52 — Enumeration: Colour model type – Example enumerators**

| <b>Name</b>    | <b>Description</b>  | <b>RL enumerator</b> |
|----------------|---|----------------------|
| CMT_Grey       | The colour type is grey scale, and 1 colour value is provided.  | “exemplary”          |
| CMT_RGB        | The colour type is red, green, blue and 3 colour values are provided.   | “exemplary”          |
| CMT_HSV        | The colour type is hue, saturation, value and 3 colour values are provided.   | “exemplary”          |
| CMT_LUV        | The colour type is luminance and colour coordinates U, V and 3 colour values are provided.  | “exemplary”          |
| CMT_RGBIR      | The colour type is red, green, blue, infrared and 4 colour values are provided.   | “exemplary”          |
| CMT_ColourList | <p>Each value references a predefined colour and 1 colour value for reference values is provided.</p> <p>Requirement:</p> <p>By using the enumerator “CMT_ColourList” the signal’s “Colour values” (see A.2.100, A.3.12, A.4.29) values are defined during the system design phase.</p> | “exemplary”          |

**A.1.15 Radial velocity ambiguity domain {begin, end}****Table A.53 — Signal: Radial velocity ambiguity domain {begin, end}**

|                    |   |             |            |
|--------------------|---|-------------|------------|
| <b>Name</b>        | Radial velocity ambiguity domain {begin, end}   |             |            |
| <b>Description</b> | The signal “Radial velocity ambiguity domain {begin, end}” (A.1.15) provides a description of the Doppler ambiguity caused by under-sampling. |             |            |
| <b>Value type</b>  | 2D vector real value  | <b>Unit</b> | (m/s, m/s) |

**A.1.16 Range ambiguity domain {begin, end}****Table A.54 — Signal: Range ambiguity domain {begin, end}**

|                    |   |             |        |
|--------------------|---|-------------|--------|
| <b>Name</b>        | Range ambiguity domain {begin, end}   |             |        |
| <b>Description</b> | The range of the ambiguity domain is defined by {begin} and {end}. The defined range is only one possible solution for an ambiguous measurement method (for example, under-sampling). |             |        |
| <b>Value type</b>  | 2D vector real value  | <b>Unit</b> | (m, m) |

**A.1.17 Angle azimuth ambiguity domain {begin, end}****Table A.55 — Signal: Angle azimuth ambiguity domain {begin, end}**

|                    |   |             |            |
|--------------------|---|-------------|------------|
| <b>Name</b>        | Angle azimuth ambiguity domain {begin, end}   |             |            |
| <b>Description</b> | The azimuth angle of the ambiguity domain is defined by {begin} and {end}. The defined azimuth angle is only one possible solution for an ambiguous measurement method. |             |            |
| <b>Value type</b>  | 2D vector real value  | <b>Unit</b> | (rad, rad) |

#### A.1.18 Angle elevation ambiguity domain {begin, end}

**Table A.56 — Signal: Angle elevation ambiguity domain {begin, end}**

|                    |   |             |            |
|--------------------|---|-------------|------------|
| <b>Name</b>        | Angle elevation ambiguity domain {begin, end}   |             |            |
| <b>Description</b> | The elevation angle of the ambiguity domain is defined by {begin} and {end}. The defined elevation angle is only one possible solution for an ambiguous measurement method. |             |            |
| <b>Value type</b>  | 2D vector real value  | <b>Unit</b> | (rad, rad) |

#### A.1.19 Interface applicability

**Table A.57 — Signal: Interface applicability**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Interface applicability  |             |   |
| <b>Description</b> | <p>The signal “Interface applicability” (A.1.19) defines the interface(s) for which the message is relevant for.</p> <p>The implementation of the interface may provide several interfaces with different signal’s “Interface applicability” (A.1.19) enumerators.</p> <p>Additional information:</p> <p>The signal “Interface applicability” (A.1.19) is necessary, for example, for a service-oriented architecture.</p> |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.58 — Enumeration: Interface applicability – Example enumerators**

| <b>Name</b>  | <b>Description</b>   | <b>RL enumerator</b> |
|--------------|--|----------------------|
| IA_OLI       | The applicability of the interface is for all OLIs.                | “exemplary”          |
| IA_FLI       | The applicability of the interface is for the FLI.                 | “exemplary”          |
| IA_DLI       | The applicability of the interface is for the DLI.                 | “exemplary”          |
| IA_FLIAndDLI | The applicability of the interface is for FLI and DLI.             | “exemplary”          |
| IA_PMOI      | The applicability of the interface is for the PMOI.                | “exemplary”          |
| IA_RDOI      | The applicability of the interface is for the RDOI.                | “exemplary”          |
| IA_SOI       | The applicability of the interface is for the SOI.                 | “exemplary”          |
| IA_FSAOI     | The applicability of the interface is for the FSAOI.               | “exemplary”          |
| IA_All       | The applicability of the interface is for all provided interfaces. | “exemplary”          |

#### A.1.20 Vehicle coordinate system type – header

**Table A.59 — Signal: Vehicle coordinate system type – header**

|             |   |
|-------------|---|
| <b>Name</b> | Vehicle coordinate system type – header |
|-------------|---|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | The signal "Vehicle coordinate system type – header" (A.1.20) defines the reference vehicle coordinate system for the interfaces of the sensor cluster. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.60 — Enumeration: Vehicle coordinate system type – header**

| <b>Name</b>    | <b>Description</b>  | <b>RL enumerator</b> |
|----------------|---|----------------------|
| VCST_RearAxle  | Use definition "vehicle rear-axle coordinate system" (3.7.12) for the ego-vehicle coordinate system of the sensor or sensor cluster.  | Mandatory            |
| VCST_RoadLevel | Use definition "vehicle road-level coordinate system" (3.7.15) for the ego-vehicle coordinate system of the sensor or sensor cluster. | Mandatory            |

**A.1.21 Sensor origin point {x, y, z} – header****Table A.61 — Signal: Sensor origin point {x, y, z} – header**

|                    |   |             |           |
|--------------------|---|-------------|-----------|
| <b>Name</b>        | Sensor origin point {x, y, z} – header  |             |           |
| <b>Description</b> | <p>The signal "Sensor origin point {x, y, z} – header" (A.1.21) provides the position of the origin of the sensing element's sensor coordinate system in the ego-vehicle coordinate system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p> <p>Additional information:<br/>The signal "Sensor origin point – correction {x, y, z}" (A.5.43) is already applied.</p> |             |           |
| <b>Value type</b>  | 3D vector real value  | <b>Unit</b> | (m, m, m) |

**A.1.22 Sensor origin point {x, y, z} – error****Table A.62 — Signal: Sensor origin point {x, y, z} – error**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Sensor origin point {x, y, z} – error  |             |   |
| <b>Description</b> | <p>The uncertainty of the sensor origin point calibration should be included in the interface {x, y, z}.</p> <p>Additional information:<br/>If only the trueness error [see "Error model implementation" (B.4.1) and "Multiple error value types" (B.4.1.3.2)] in the overall system is known, the sensor origin point error shall not be sent.<br/>No high dynamic motion of the ego-vehicle is compensated by the signal.<br/>The signal "Sensor origin point – correction {x, y, z} – error" (A.5.44) is already applied.</p> |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal "Sensor origin point {x, y, z} – header" (A.1.21). |

**A.1.23 Sensor orientation {yaw, pitch, roll} – header****Table A.63 — Signal: Sensor orientation {yaw, pitch, roll} – header**

|             |  |
|-------------|--|
| <b>Name</b> | Sensor orientation {yaw, pitch, roll} – header |
|-------------|--|

|                    |   |             |                 |
|--------------------|---|-------------|-----------------|
| <b>Description</b> | <p>The signal “Sensor orientation {yaw, pitch, roll} – header” (A.1.23) provides the relative orientation of the sensor axis system with respect to the ego-vehicle axis system [see signal “Vehicle coordinate system type – header” (A.1.20)].</p> <p>The sensor orientation represents the current orientation of the sensor origin point to the best knowledge. It includes the pose estimation given by the calibration. The uncertainty of this estimation is given with the corresponding error. The estimation does not include effects due to short-time dynamics, such as pitch angle changes during braking, but can include long-time effects resulting from, for example, luggage in the trunk.</p> <p>Additional information:</p> <p>The signal “Sensor orientation {yaw, pitch, roll} – header” (A.1.23) is given with {yaw, pitch, roll} angles, which have to be separated in a defined procedure.</p> <p>For example, the rotations are to be performed yaw first (around the Z-axis), pitch second (around the new Y-axis) and roll third (around the new X-axis) following the definition according to ISO 8855:2011.</p> <p>The signal “Sensor orientation – correction {yaw, pitch, roll}” (A.5.46) is already applied.</p> |             |                 |
| <b>Value type</b>  | 3D vector real value  | <b>Unit</b> | (rad, rad, rad) |

#### A.1.24 Sensor orientation {yaw, pitch, roll} – error

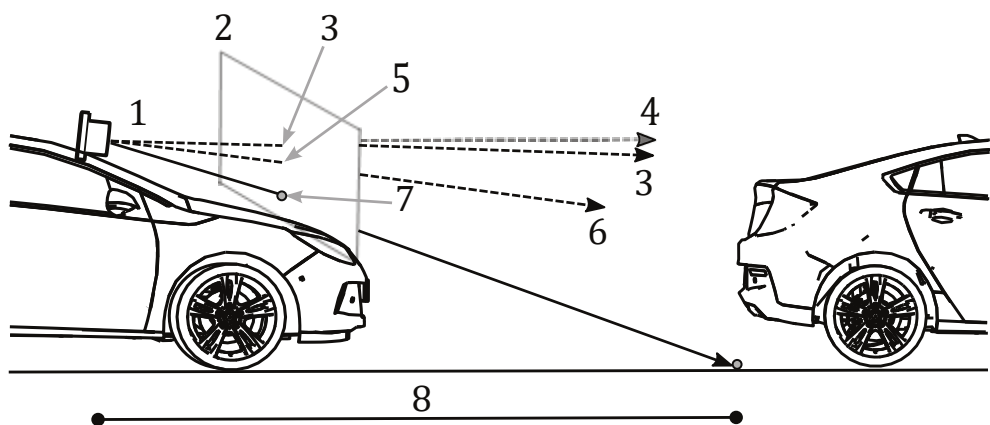
**Table A.64 — Signal: Sensor orientation {yaw, pitch, roll} – error**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Sensor orientation {yaw, pitch, roll} – error  |             |   |
| <b>Description</b> | <p>The uncertainty of the sensor orientation calibration should be included in the interface {yaw, pitch, roll}.</p> <p>Additional information:</p> <p>If only the trueness error in the overall system is known, the sensor orientation error shall not be sent.</p> <p>No high dynamic motion of the ego-vehicle is compensated by the signal.</p> <p>The signal “Sensor orientation – correction {yaw, pitch, roll} – error” (A.5.47) is already applied.</p> |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Sensor orientation {yaw, pitch, roll} – header” (A.1.23). |

#### A.1.25 Vanishing point {azimuth, elevation}

**Table A.65 — Signal: Vanishing point {azimuth, elevation}**

|                    |   |             |            |
|--------------------|---|-------------|------------|
| <b>Name</b>        | Vanishing point {azimuth, elevation}  |             |            |
| <b>Description</b> | <p>A set of lines in the image plane that corresponds to a set of parallel surface lines in the 3D world space converges to a common point in the image space, known as the signal “Vanishing point {azimuth, elevation}” (A.1.25).</p> <p>Additional information:</p> <p>Figure A.1 shows the distance estimation using the vanishing point.</p> |             |            |
| <b>Value type</b>  | 2D vector real value  | <b>Unit</b> | (rad, rad) |



- Key**
- 1 camera sensor
  - 2 image plane (enlarged)
  - 3 vanishing point
  - 4 ego-vehicle X-axis direction
  - 5 image centre of the camera sensor
  - 6 optical axis of the camera
  - 7 image of a point on road level of the camera sensor
  - 8 distance of a recognised entity (in X-axis direction of the ego-vehicle coordinate system)

Figure A.1 — Example for distance estimation

A.1.26Vanishing point {azimuth, elevation} – error

Table A.66 — Signal: Vanishing point {azimuth, elevation} – error

|             |  |      |  |
|-------------|--|------|--|
| Name        | Vanishing point {azimuth, elevation} – error   |      |  |
| Description | The signal “Vanishing point {azimuth, elevation} – error” (A.1.26) provides the uncertainty of the signal “Vanishing point {azimuth, elevation}” (A.1.25). |      |  |
| Value type  | scalar/vector/matrix value (see B.4.1)   | real | Unit See signal “Vanishing point {azimuth, elevation}” (A.1.25). |

A.1.27Sender ID

Table A.67 — Signal: Sender ID

|             |  |      |   |
|-------------|--|------|---|
| Name        | Sender ID  |      |   |
| Description | The signal “Sender ID” (A.1.27) uniquely identifies the sender (for example, an ECU) and connects the data of different senders’ interfaces and merges the data. |      |   |
| Value type  | integer value  | Unit | 1 |

A.2 Object level entity signals

Signals are defined for the entities of subclauses 7.3-7.6 in this subclause (see Table A.68-Table A.336). These interfaces are located at OLI.

Normally, the OLI originates from several sensors. Therefore, at object level, the term sensor cluster is always used, even if a single sensor is serving the interface.

### A.2.1 Existence probability – object level

**Table A.68 — Signal: Existence probability – object level**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Existence probability – object level  |             |   |
| <b>Description</b> | <p>The signal “Existence probability – object level” (A.2.1) represents the recognition quality and sensor’s certainty that the object exists. A higher value indicates an object which is more certain to exist while a lower value indicates a potential false positive object.</p> <p>Additional information:</p> <p>Low values may be possible in the first cycles of the object’s appearance.</p> <p>Use as confidence measure where a low value means less confidence and a high value indicates strong confidence.</p> |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

### A.2.2 Object ID

**Table A.69 — Signal: Object ID**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Object ID   |             |   |
| <b>Description</b> | <p>The signal “Object ID” (A.2.2) provides the unique ID of the object within the object lists of the sensor cluster’s interfaces.</p> <p>Additional information:</p> <p>The signal’s “Object ID” (A.2.2) value is only unique for each sensor cluster’s recognised entities of its OLI.</p> <p>The signal’s “Object ID” (A.2.2) value is unique for the same entity in multiple cycles.</p> <p>Reuse of the signal’s “Object ID” (A.2.2) value is possible under defined conditions.</p> <p>Requirement:</p> <p>The reuse of the signal’s “Object ID” (A.2.2) value shall be well-defined.</p> |             |   |
| <b>Value type</b>  | integer value   | <b>Unit</b> | 1 |

### A.2.3 Object grouping ID

**Table A.70 — Signal: Object grouping ID**

|             |                    |  |  |
|-------------|--------------------|--|--|
| <b>Name</b> | Object grouping ID |  |  |
|-------------|--------------------|--|--|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | <p>A sensor cluster can group objects from the same object interface or different OLIs of the sensor cluster which are linked together. All objects from one real-world entity should have the same signal's "Object grouping ID" (A.2.3) value. The ID is unique considering all OLIs of a sensor cluster. In cases in which it is not clear, if it is, for example, a single traffic participant or multiple, the link can be used.</p> <p>EXAMPLE 1 Long vehicles (for example, a truck with a trailer) can be described by two potentially moving objects. In this case the same value for the signal "Object grouping ID" (A.2.3) logically connects or links one object with the other object.</p> <p>EXAMPLE 2 A construction site vehicle (as a potentially moving object) with a traffic sign (as static object) at the rear of the vehicle.</p> <p>Additional information:<br/>The signal "Object grouping ID" (A.2.3) := 0 if object is not associated with another object.</p> <p>Requirement:<br/>The reuse of the signal's "Object grouping ID" (A.2.3) values shall be well defined.</p> |             |   |
| <b>Value type</b>  | integer value   | <b>Unit</b> | 1 |

#### A.2.4 Age

**Table A.71 — Signal: Age**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Age   |             |   |
| <b>Description</b> | <p>The signal "Age" (A.2.4) provides the amount of time that this object has been currently tracked.</p> <p>Additional information:<br/>The signal "Age" (A.2.4) := current time stamp – time stamp when the tracking or prediction for this object has been initiated.</p> |             |   |
| <b>Value type</b>  | [0...] real value   | <b>Unit</b> | s |

#### A.2.5 Number of valid observations – object level

**Table A.72 — Signal: Number of valid observations – object level**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid observations – object level  |             |   |
| <b>Description</b> | <p>The signal "Number of valid observations – object level" (A.2.5) provides the current number of valid tuples [specifically the signal "Time stamp reference – object level" (A.2.6) and "Observation status – object level" (A.2.7)].</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

#### A.2.6 Time stamp reference – object level

**Table A.73 — Signal: Time stamp reference – object level**

|                    |   |  |  |
|--------------------|---|--|--|
| <b>Name</b>        | Time stamp reference – object level   |  |  |
| <b>Description</b> | <p>The signal "Time stamp reference – object level" (A.2.6) provides a reference to a previously sent interface message with the referenced time stamp for signal "Time stamp – &lt;...&gt;" (A.1.5).</p> |  |  |



|                   |            |             |   |
|-------------------|------------|-------------|---|
| <b>Value type</b> | real value | <b>Unit</b> | s |
|-------------------|------------|-------------|---|

### A.2.7 Observation status – object level

**Table A.74 — Signal: Observation status – object level**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Observation status – object level   |             |   |
| <b>Description</b> | The signal “Observation status – object level” (A.2.7) provides the observation status of the object, which was recognised in a previous cycle [see signal “Time stamp reference – object level” (A.2.6) which references the cycle]. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.75 — Enumeration: Observation status – object level – Example enumerators**

| <b>Name</b> | <b>Description</b>  | <b>RL enumerator</b> |
|-------------|---|----------------------|
| OS_True     | The object was observed in the referenced cycle.  | “exemplary”          |
| OS_False    | The object was not observed in the referenced cycle. It may be predicted in the referenced cycle. | “exemplary”          |

### A.2.8 Track quality

**Table A.76 — Signal: Track quality**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Track quality   |             |   |
| <b>Description</b> | The signal “Track quality” (A.2.8) value describes how well the track was predicted of the current observation. Usually there is a variety of signals which are used to set up a track. The quality of the track is changing gradually from a precise prediction, where all values met the prediction very close, down to a prediction which is based on less signals and where the deviation gets large. A value close to 100 % indicates that both the prediction is very precise and most of available signals are used to build the track. A very low value (near 0 %) means a very high deviation of the observation from the predicted track and that the availability of the signals is lower. |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

### A.2.9 Measurement status – object level

**Table A.77 — Signal: Measurement status – object level**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Measurement status – object level  |             |   |
| <b>Description</b> | The signal “Measurement status – object level” (A.2.9) provides the current measurement status of the object. The objects should be sent out as fast as possible, even if not all statuses are observed, and the object’s signals might not be fully filled. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.78 — Enumeration: Measurement status – object level – Example enumerators**

| <b>Name</b> | <b>Description</b>  | <b>RL enumerator</b> |
|-------------|---|----------------------|
| MS_Measured | The object was updated with new information from the sensor in the last update cycle of this object list. | “exemplary”          |
| MS_New      | This indicates the first occurrence of this particular object in the object list.                         | “exemplary”          |

|                      |  |             |
|----------------------|--|-------------|
| MS_Predicted         | The object was predicted without any new information with respect to the previous update cycle.  | “exemplary” |
| MS_PartlyMeasured    | The object was partly updated with new information from the sensor in the last update cycle of this object list. The measurement or tracking was incomplete and could, therefore, only partially update the signals of this object. The remaining signals of this object are unchanged or predicted for this update cycle. | “exemplary” |
| MS_Invalid           | The object is invalid in this cycle.   | “exemplary” |
| MS_PredictedOccluded | Tracked object is temporarily occluded by another entity.  | “exemplary” |

#### A.2.10 Tracking motion model

**Table A.79 — Signal: Tracking motion model**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Tracking motion model   |             |   |
| <b>Description</b> | The signal “Tracking motion model” (A.2.10) provides the used motion model for tracking moving items. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.80 — Enumeration: Tracking motion model – Example enumerators**

| <b>Name</b>              | <b>Description</b>                            | <b>RL enumerator</b> |
|--------------------------|---|----------------------|
| TMM_ConstantVelocity     | The motion model uses constant velocity.      | “exemplary”          |
| TMM_ConstantAcceleration | The motion model uses constant acceleration.  | “exemplary”          |
| TMM_ConstantRotationRate | The motion model uses constant rotation rate. | “exemplary”          |

#### A.2.11 Number of valid potentially moving object classifications

**Table A.81 — Signal: Number of valid potentially moving object classifications**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid potentially moving object classifications  |             |   |
| <b>Description</b> | The signal “Number of valid potentially moving object classifications” (A.2.11) provides the current number of valid tuples [specifically the signal “Potentially moving object classification type” (A.2.12) and the signal “Potentially moving object classification type – confidence” (A.2.13)] for classifying a potentially moving object. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

#### A.2.12 Potentially moving object classification type

**Table A.82 — Signal: Potentially moving object classification type**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Potentially moving object classification type   |             |   |
| <b>Description</b> | The signal “Potentially moving object classification type” (A.2.12) provides a list of classification types for potentially moving objects. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.83 — Enumeration: Potentially moving object classification type – Example enumerators**

| <b>Name</b> | <b>Description</b> | <b>RL enumerator</b> |
|-------------|--------------------|----------------------|
|-------------|--------------------|----------------------|

|                         |  |             |
|-------------------------|--|-------------|
| PMOCT_Car               | The potential moving object represents an object car.  | "exemplary" |
| PMOCT_Truck             | The potential moving object represents an object truck.  | "exemplary" |
| PMOCT_HeavyTruck        | The potential moving object represents an object heavy truck.                                    | "exemplary" |
| PMOCT_Van               | The potential moving object represents an object van.  | "exemplary" |
| PMOCT_Bus               | The potential moving object represents an object bus.  | "exemplary" |
| PMOCT_Trailer           | The potential moving object represents an object trailer.  | "exemplary" |
| PMOCT_Train             | The potential moving object represents an object train.  | "exemplary" |
| PMOCT_OtherVehicle      | The potential moving object represents an object unidentified vehicle.                           | "exemplary" |
| PMOCT_Motorbike         | The potential moving object represents an object motorbike.                                      | "exemplary" |
| PMOCT_Animal            | The potential moving object represents an object animal.   | "exemplary" |
| PMOCT_Bicycle           | The potential moving object represents an object bicycle.  | "exemplary" |
| PMOCT_Tricycle          | The potential moving object represents an object tricycle.                                       | "exemplary" |
| PMOCT_Pedestrian        | The potential moving object represents an object pedestrian.                                     | "exemplary" |
| PMOCT_Wheelchair        | The potential moving object represents an object wheelchair.                                     | "exemplary" |
| PMOCT_Ambulance         | The potential moving object represents an object ambulance.                                      | "exemplary" |
| PMOCT_FireEngine        | The potential moving object represents an object fire engine.                                    | "exemplary" |
| PMOCT_PoliceCar         | The potential moving object represents an object police car.                                     | "exemplary" |
| PMOCT_SanitationVehicle | The potential moving object represents an object sanitation vehicle.                             | "exemplary" |
| PMOCT_UtilityVehicle    | The potential moving object represents an object utility vehicle.                                | "exemplary" |
| PMOCT_TrafficController | The potential moving object represents an object traffic controller (traffic authorised person). | "exemplary" |

### A.2.13 Potentially moving object classification type – confidence

**Table A.84 — Signal: Potentially moving object classification type – confidence**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Potentially moving object classification type – confidence   |             |   |
| <b>Description</b> | The signal "Potentially moving object classification type – confidence" (A.2.13) provides the certainty for the corresponding signal "Potentially moving object classification type" (A.2.12). |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

### A.2.14 Reference point

**Table A.85 — Signal: Reference point**

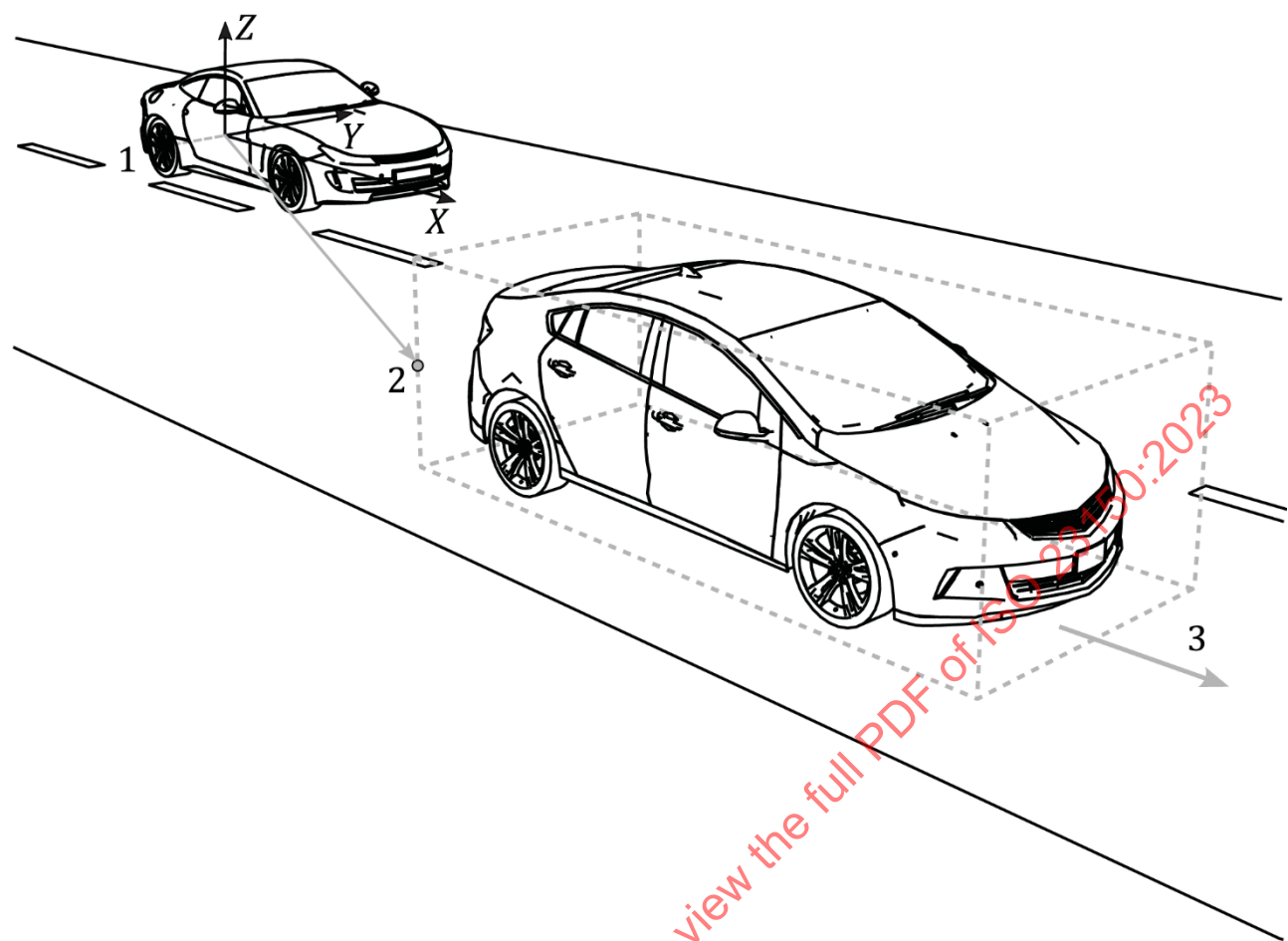
|             |                 |
|-------------|-----------------|
| <b>Name</b> | Reference point |
|-------------|-----------------|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | <p>The signal “Reference point” (A.2.14) provides the outer edges of the recognised object’s bounding box (see Figure A.2). During tracking, this point can change in each cycle.</p> <p>Additional information:</p> <p>Only valid for cuboid model. The reference point is always on the outer edge with z value “middle” (middle height), if it is not defined in the enumeration. “middle” is the geometrical middle of the bounding box for each axis.</p> <p>If the signal is not part of the implemented interface, the reference point is located in the midside, midwidth, midheight of the object’s bounding box (see enumerator “RP_MidsideMidwidthMidheight”).</p> <p>Front/rear, left/right, top/bottom are defined for the bounding box with respect to the object’s orientation [see signal “Orientation {yaw, pitch, roll}” (A.2.17)].</p> <p>Naming for enumerators:</p> <ul style="list-style-type: none"> <li>- X-axis of the bounding box: Rear – Midside – Front;</li> <li>- Y-axis of the bounding box: Right – Midwidth – Left;</li> <li>- Z-axis of the bounding box: Bottom – Midheight – Top.</li> </ul> |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.86 — Enumeration: Reference point – Example enumerators**

| Name                      | Description  | RL enumerator |
|---------------------------|--|---------------|
| RP_FrontLeftTop           | The reference point of the object’s bounding box is located at the front, left, top corner of the bounding box.        | “exemplary”   |
| RP_FrontMidwidthTop       | The reference point of the object’s bounding box is located in the middle of the front, top edge of the bounding box.  | “exemplary”   |
| RP_FrontRightTop          | The reference point of the object’s bounding box is located at the front, right, top corner of the bounding box.       | “exemplary”   |
| RP_MidsideLeftTop         | The reference point of the object’s bounding box is located in the middle of the left, top edge of the bounding box.   | “exemplary”   |
| RP_MidsideMidwidthTop     | The reference point of the object’s bounding box is located at the geometric centre of the bounding box’s top plane.   | “exemplary”   |
| RP_MidsideRightTop        | The reference point of the object’s bounding box is located in the middle of the right, top edge of the bounding box.  | “exemplary”   |
| RP_RearLeftTop            | The reference point of the object’s bounding box is located at the rear, left, top corner of the bounding box.         | “exemplary”   |
| RP_RearMidwidthTop        | The reference point of the object’s bounding box is located in the middle of the rear, top edge of the bounding box.   | “exemplary”   |
| RP_RearRightTop           | The reference point of the object’s bounding box is located at the rear, right, top corner of the bounding box.        | “exemplary”   |
| RP_FrontLeftMidheight     | The reference point of the object’s bounding box is located in the middle of the front, left edge of the bounding box. | “exemplary”   |
| RP_FrontMidwidthMidheight | The reference point of the object’s bounding box is located at the geometric centre of the bounding box’s front plane. | “exemplary”   |

| Name                        | Description  | RL enumerator |
|-----------------------------|--|---------------|
| RP_FrontRightMidheight      | The reference point of the object's bounding box is located in the middle of the front, right edge of the bounding box.    | "exemplary"   |
| RP_MidsideLeftMidheight     | The reference point of the object's bounding box is located at the geometric centre of the bounding box's left-side plane. | "exemplary"   |
| RP_MidsideMidwidthMidheight | The reference point of the object's bounding box is located at the geometric centre of the bounding box.                   | "exemplary"   |
| RP_MidsideRightMidheight    | The reference point is located at the geometric centre of the bounding box's right-side plane.                             | "exemplary"   |
| RP_RearLeftMidheight        | The reference point of the object's bounding box is located in the middle of the rear, left edge of the bounding box.      | "exemplary"   |
| RP_RearMidwidthMidheight    | The reference point of the object's bounding box is located at the geometric centre of the bounding box's rear plane.      | "exemplary"   |
| RP_RearRightMidheight       | The reference point of the object's bounding box is located in the middle of the rear, right edge of the bounding box.     | "exemplary"   |
| RP_FrontLeftBottom          | The reference point of the object's bounding box is located at the front, left, bottom corner of the bounding box.         | "exemplary"   |
| RP_FrontMidwidthBottom      | The reference point of the object's bounding box is located in the middle of the front, bottom edge of the bounding box.   | "exemplary"   |
| RP_FrontRightBottom         | The reference point of the object's bounding box is located at the front, right, bottom corner of the bounding box.        | "exemplary"   |
| RP_MidsideLeftBottom        | The reference point of the object's bounding box is located in the middle of the left, bottom edge of the bounding box.    | "exemplary"   |
| RP_MidsideMidwidthBottom    | The reference point of the object's bounding box is located at the geometric centre of the bounding box's bottom plane.    | "exemplary"   |
| RP_MidsideRightBottom       | The reference point of the object's bounding box is located in the middle of the right, bottom edge of the bounding box.   | "exemplary"   |
| RP_RearLeftBottom           | The reference point of the object's bounding box is located at the rear, left, bottom corner of the bounding box.          | "exemplary"   |
| RP_RearMidwidthBottom       | The reference point of the object's bounding box is located in the middle of the rear, bottom edge of the bounding box.    | "exemplary"   |
| RP_RearRightBottom          | The reference point of the object's bounding box is located at the rear, right, bottom corner of the bounding box.         | "exemplary"   |



- Key**
- 1 vehicle's rear axle with ego-vehicle coordinate system
  - 2 point of signal "Reference point" (A.2.14) with enumerator "RP\_RearRightMidheight" refers to rear right midheight of the potentially moving object's bounding box and orientation at this point [see signal "Orientation {yaw, pitch, roll}" (A.2.17) and signal "Position – object level {x, y, z}" (A.2.15)]
  - 3 potentially moving object's moving direction

**Figure A.2 — Example for reference point of a potentially moving object's bounding box**

**A.2.15Position – object level {x, y, z}**

**Table A.87 — Signal: Position – object level {x, y, z}**

| Name        | Position – object level {x, y, z}  |      |           |
|-------------|--|------|-----------|
| Description | <p>The signal "Position – object level {x, y, z}" (A.2.15) provides a position of the object. It is defined in the ego-vehicle coordinate system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p> <p>Additional information:<br/>The origin point of the object coordinate system is the ego-vehicle coordinate system (see Figure A.2).</p> |      |           |
| Value type  | 3D vector real value   | Unit | (m, m, m) |

**A.2.16 Position – object level {x, y, z} – error****Table A.88 — Signal: Position – object level {x, y, z} – error**

|                    |   |             |  |
|--------------------|---|-------------|--|
| <b>Name</b>        | Position – object level {x, y, z} – error   |             |  |
| <b>Description</b> | The signal “Position – object level {x, y, z} – error” (A.2.16) provides the error which represents the uncertainty of the state estimation of the signal’s “Position – object level {x, y, z}” (A.2.15) position (see Figure A.2). |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal “Position – object level {x, y, z}” (A.2.15). |

**A.2.17 Orientation {yaw, pitch, roll}****Table A.89 — Signal: Orientation {yaw, pitch, roll}**

|                    |  |             |                 |
|--------------------|--|-------------|-----------------|
| <b>Name</b>        | Orientation {yaw, pitch, roll}   |             |                 |
| <b>Description</b> | <p>The signal “Orientation {yaw, pitch, roll}” (A.2.17) provides the orientation of the object’s bounding box defined in the ego-vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)].</p> <p>Additional information:</p> <p>A potentially moving object may move in another direction as the X-axis of the bounding box [see signal “Bounding box extent {length, width, height}” (A.2.20)].</p> <p>The axis system of the bounding box and the object’s axis system may be different.</p> <p>The signal “Orientation {yaw, pitch, roll}” (A.2.17) is given with {yaw, pitch, roll} angles, which have to be separated in a defined procedure.</p> <p>For example, the rotations are to be performed yaw first (around the Z-axis), pitch second (around the new Y-axis) and roll third (around the new X-axis) following the definition according to ISO 8855:2011.</p> |             |                 |
| <b>Value type</b>  | 3D vector real value   | <b>Unit</b> | (rad, rad, rad) |

**A.2.18 Orientation {yaw, pitch, roll} – error****Table A.90 — Signal: Orientation {yaw, pitch, roll} – error**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Orientation {yaw, pitch, roll} – error  |             |   |
| <b>Description</b> | The signal “Orientation {yaw, pitch, roll} – error” (A.2.18) provides the error which represents the uncertainty of the state estimation of the signal’s “Orientation {yaw, pitch, roll}” (A.2.17) values for orientation (see Figure A.2). |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal “Orientation {yaw, pitch, roll}” (A.2.17). |

**A.2.19 Road level****Table A.91 — Signal: Road level**

|             |            |
|-------------|------------|
| <b>Name</b> | Road level |
|-------------|------------|



|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | <p>The signal “Road level” (A.2.19) provides the information about relevance to ego-vehicle road level, for example, in case no {z} value for position can be given [see signal “Position – object level {x, y, z}” (A.2.15)].</p> <p>Additional information:</p> <p>In situations it happens that a target vehicle is detected by the sensor, but the vehicle is on different road levels compared to the ego-vehicle (for example, motorway bridge ahead on which a vehicle is crossing the motorway). This signal indicates the relevance to the ego-vehicle road level.</p> |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

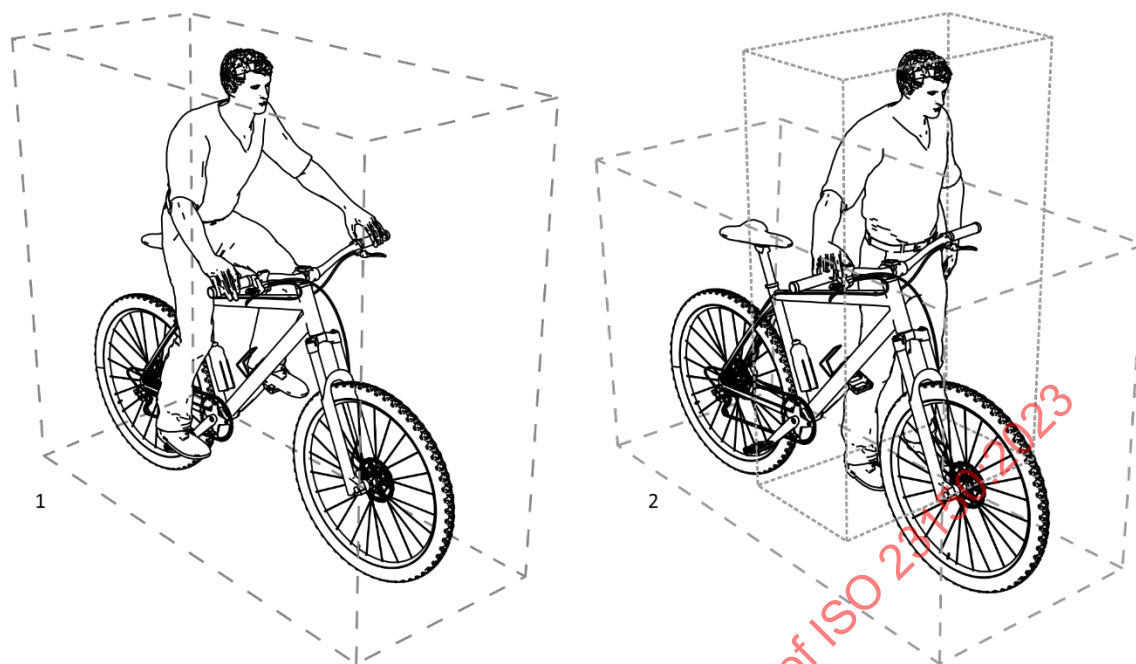
**Table A.92 — Enumeration: Road level – Example enumerators**

| Name              | Description  | RL enumerator |
|-------------------|--|---------------|
| RL_EgoRoadLevel   | The object is on the same road level as the ego-vehicle.                                 | “exemplary”   |
| RL_RoadLevelAbove | The object is on a different road level as the ego-vehicle. It is on a road level above. | “exemplary”   |
| RL_RoadLevelBelow | The object is on a different road level as the ego-vehicle. It is on a road level below. | “exemplary”   |

**A.2.20 Bounding box extent {length, width, height}****Table A.93 — Signal: Bounding box extent {length, width, height}**

|                    |   |             |           |
|--------------------|---|-------------|-----------|
| <b>Name</b>        | Bounding box extent {length, width, height}   |             |           |
| <b>Description</b> | <p>The signal “Bounding box extent {length, width, height}” (A.2.20) provides the extent of an object (see Figure A.3).</p> <p>{length}: distance between the rear plane and the front plane of the bounding box. It is the object dimension along the X-axis of the bounding box axis system (see Table A.86).</p> <p>{width}: distance between the right-side plane and the left-side plane of the bounding box. It is the object dimension along the Y-axis of the bounding box axis system (see Table A.86).</p> <p>{height}: distance between the bottom plane and the top plane of the bounding box. It is the object dimension along the Z-axis of the bounding box axis system (see Table A.86).</p> <p>Additional information:</p> <p>The orientation of the bounding box is defined by the signal “Orientation {yaw, pitch, roll}” (A.2.17).</p> <p>Requirement:</p> <p>The degree to which small structures are part of the extent of the bounding box shall be defined by the sensor manufacturer during the system design phase [see signal “Included geometric structures” (A.2.23)].</p> |             |           |
| <b>Value type</b>  | 3D vector real value  | <b>Unit</b> | (m, m, m) |



**Key**

- 1 one bounding box of one potentially moving object
- 2 two bounding boxes of two potentially moving object which are connected [see signal "Accessory connection to person type" (A.2.62)]

**Figure A.3 — Example for the signal "Bounding box extent {length, width, height}" (A.2.20)**

#### A.2.21 Bounding box extent {length, width, height} - error

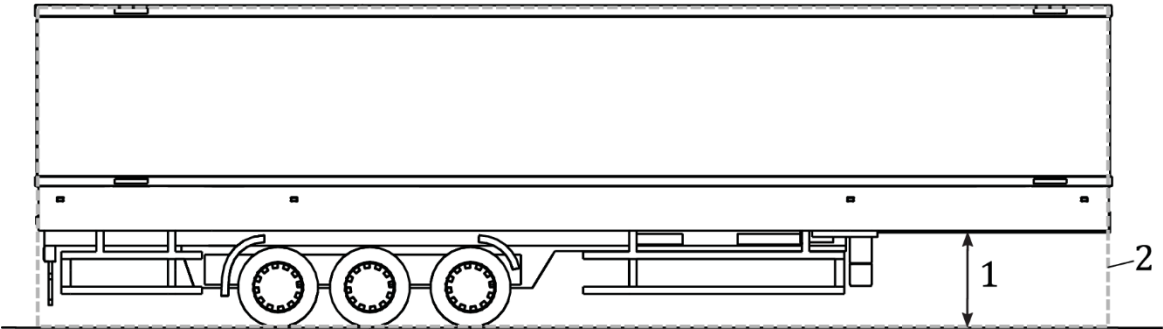
**Table A.94 — Signal: Bounding box extent {length, width, height} - error**

|                    |  |             |  |
|--------------------|--|-------------|--|
| <b>Name</b>        | Bounding box extent {length, width, height} - error  |             |  |
| <b>Description</b> | The signal "Bounding box extent {length, width, height} - error" (A.2.21) provides the error of the signal "Bounding box extent {length, width, height}" (A.2.20). |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal "Bounding box extent {length, width, height}" (A.2.20). |

#### A.2.22 Clearance of the object {height}

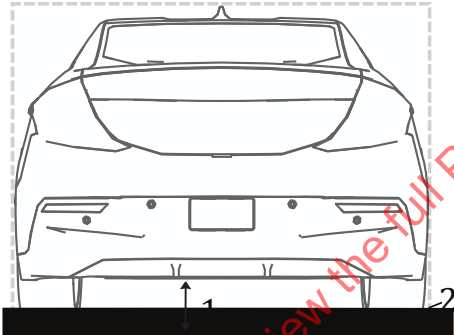
**Table A.95 — Signal: Clearance of the object {height}**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Clearance of the object {height}   |             |   |
| <b>Description</b> | The signal "Clearance of the object {height}" (A.2.22) provides the maximum height of the clearance area between ground and object lower shape. The bounding box is indicated with the dashed box (see Figure A.4 and Figure A.5). |             |   |
| <b>Value type</b>  | [0...] real value  | <b>Unit</b> | m |



- Key**
- 1 signal “Clearance of the object {height}” (A.2.22)
  - 2 bounding box

**Figure A.4 — Example for the signal “Clearance of the object {height}” (A.2.22) of a trailer**



- Key**
- 1 signal “Clearance of the object {height}” (A.2.22)
  - 2 bounding box

**Figure A.5 — Example for the signal “Clearance of the object {height}” (A.2.22) of a vehicle**

**A.2.23Included geometric structures**

**Table A.96 — Signal: Included geometric structures**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Included geometric structures  |             |   |
| <b>Description</b> | The signal “Included geometric structures” (A.2.23) provides the geometrical structures that are taken into account in the bounding box [see signal “Bounding box extent {length, width, height}” (A.2.20)]. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

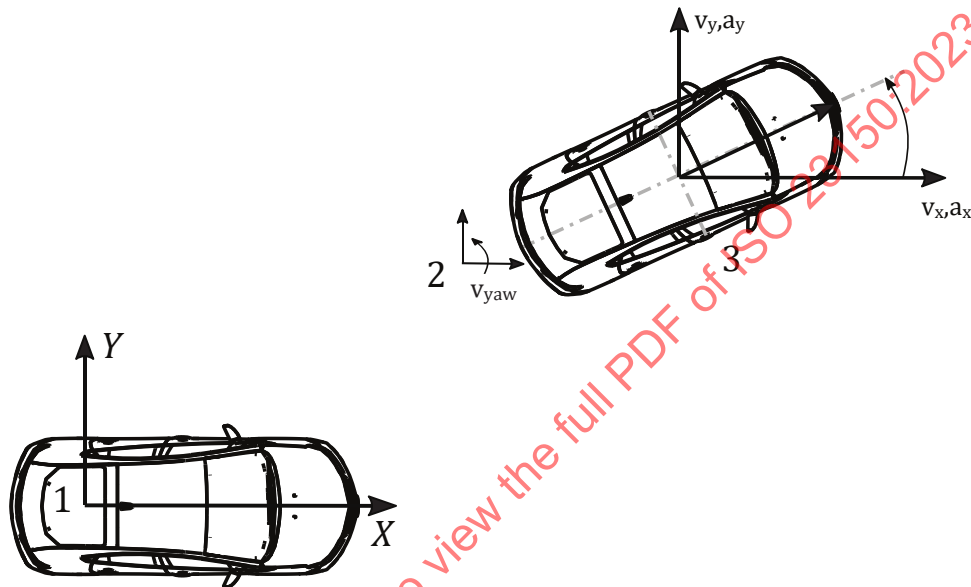
**Table A.97 — Enumeration: Included geometric structures - Example enumerators**

|                    |  |                      |
|--------------------|--|----------------------|
| <b>Name</b>        | <b>Description</b>   | <b>RL enumerator</b> |
| IGS_WithoutMirrors | The bounding box dimensions exclude vehicle mirrors [see signal “Bounding box extent {length, width, height}” (A.2.20)]. | “exemplary”          |
| IGS_WithMirrors    | The bounding box dimensions include vehicle mirrors [see signal “Bounding box extent {length, width, height}” (A.2.20)]. | “exemplary”          |

## A.2.24 Velocity {x, y, z} – object level

Table A.98 — Signal: Velocity {x, y, z} – object level

|                    |  |             |                 |
|--------------------|--|-------------|-----------------|
| <b>Name</b>        | Velocity {x, y, z} – object level  |             |                 |
| <b>Description</b> | The signal “Velocity {x, y, z} – object level” (A.2.24) provides the velocity of the object with respect to the signal “Motion type” (A.1.13) (see Figure A.6) in the ego-vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)]. |             |                 |
| <b>Value type</b>  | 3D vector real value   | <b>Unit</b> | (m/s, m/s, m/s) |



## Key

- 1 ego-vehicle coordinate system
- 2 instantaneous centre of rotation of the potentially moving object
- 3 potentially moving object with ego-vehicle's axis system

Figure A.6 — Example for an ego-vehicle and object dynamics of a potentially moving object

## A.2.25 Velocity {x, y, z} – object level – error

Table A.99 — Signal: Velocity {x, y, z} – object level – error

|                    |  |             |  |
|--------------------|--|-------------|--|
| <b>Name</b>        | Velocity {x, y, z} – object level – error  |             |  |
| <b>Description</b> | The signal “Velocity {x, y, z} – object level – error” (A.2.25) provides the error which represents the uncertainty of the state estimation of {x, y, z} velocity with respect to the signal “Motion type” (A.1.13) (see Figure A.6) in the ego-vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)]. |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Velocity {x, y, z} – object level” (A.2.24). |

## A.2.26 Acceleration {x, y, z}

Table A.100 — Signal: Acceleration {x, y, z}

|             |                        |
|-------------|------------------------|
| <b>Name</b> | Acceleration {x, y, z} |
|-------------|------------------------|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | The signal "Acceleration {x, y, z}" (A.2.26) provides the acceleration of the object with respect to the signal "Motion type" (A.1.13) (see Figure A.6) in the ego-vehicle coordinate system [see signal "Vehicle coordinate system type – header" (A.1.20)]. |             |   |
| <b>Value type</b>  | 3D vector real value  | <b>Unit</b> | (m/s <sup>2</sup> , m/s <sup>2</sup> , m/s <sup>2</sup> ) |

#### A.2.27 Acceleration {x, y, z} – error

**Table A.101 — Signal: Acceleration {x, y, z} – error**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Acceleration {x, y, z} – error  |             |   |
| <b>Description</b> | The signal "Acceleration {x, y, z} – error" (A.2.27) provides the error which represents the uncertainty of the state estimation of {x, y, z} acceleration with respect to the signal "Motion type" (A.1.13) (see Figure A.6) in the ego-vehicle coordinate system [see signal "Vehicle coordinate system type – header" (A.1.20)]. |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal "Acceleration {x, y, z}" (A.2.26). |

#### A.2.28 Instantaneous centre of rotation {x, y}

**Table A.102 — Signal: Instantaneous centre of rotation {x, y}**

|                    |  |             |        |
|--------------------|--|-------------|--------|
| <b>Name</b>        | Instantaneous centre of rotation {x, y}  |             |        |
| <b>Description</b> | <p>The signal "Instantaneous centre of rotation {x, y}" (A.2.28) provides the centre-point of the instantaneous rotation in the ego-vehicle coordinate system [see signal "Vehicle coordinate system type – header" (A.1.20)]. The centre can be located outside the object's bounding box (see Figure A.6).</p> <p>Additional information:<br/>The centre-point can be arbitrary for the object. The centre-point is the origin of the rotation rates [see signal "Rotation rate at instantaneous centre of rotation {yaw}" (A.2.30)] and it is defined in the ego-vehicle coordinate system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p> |             |        |
| <b>Value type</b>  | 2D vector real value   | <b>Unit</b> | (m, m) |

#### A.2.29 Instantaneous centre of rotation {x, y} – error

**Table A.103 — Signal: Instantaneous centre of rotation {x, y} – error**

|                    |  |             |  |
|--------------------|--|-------------|--|
| <b>Name</b>        | Instantaneous centre of rotation {x, y} – error  |             |  |
| <b>Description</b> | The signal "Instantaneous centre of rotation {x, y} – error" (A.2.29) provides the error of the state estimation of the signal "Instantaneous centre of rotation {x, y}" (A.2.28) (see Figure A.6) in the ego-vehicle coordinate system [see signal "Vehicle coordinate system type – header" (A.1.20)]. |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal "Instantaneous centre of rotation {x, y}" (A.2.28). |

#### A.2.30 Rotation rate at instantaneous centre of rotation {yaw}

**Table A.104 — Signal: Rotation rate at instantaneous centre of rotation {yaw}**

|             |   |  |  |
|-------------|---|--|--|
| <b>Name</b> | Rotation rate at instantaneous centre of rotation {yaw} |  |  |
|-------------|---|--|--|

|                    |  |             |       |
|--------------------|--|-------------|-------|
| <b>Description</b> | The signal “Rotation rate at instantaneous centre of rotation {yaw}” (A.2.30) provides the {yaw}-rate of the object at the signal “Instantaneous centre of rotation {x, y}” (A.2.28) with respect to the signal “Motion type” (A.1.13) (see Figure A.6) and with respect to the ego-vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)]. |             |       |
| <b>Value type</b>  | 1D vector real value   | <b>Unit</b> | rad/s |

### A.2.31 Rotation rate at instantaneous centre of rotation {yaw} – error

**Table A.105 — Signal: Rotation rate at instantaneous centre of rotation {yaw} – error**

|                    |  |             |  |
|--------------------|--|-------------|--|
| <b>Name</b>        | Rotation rate at instantaneous centre of rotation {yaw} – error  |             |  |
| <b>Description</b> | The signal “Rotation rate at instantaneous centre of rotation {yaw} – error” (A.2.31) provides the error of the {yaw}-rate at the object’s signal “Instantaneous centre of rotation {x, y}” (A.2.28) with respect to the signal “Motion type” (A.1.13) (see Figure A.6) and with respect to the ego-vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)]. |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value (see B.4.1)  | <b>Unit</b> | See signal “Rotation rate at instantaneous centre of rotation {yaw}” (A.2.30). |

### A.2.32 Movement status

**Table A.106 — Signal: Movement status**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Movement status   |             |   |
| <b>Description</b> | The signal “Movement status” (A.2.32) provides the information about a possible movement of the object during tracking. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.107 — Enumeration: Movement status – Example enumerators**

| <b>Name</b>      | <b>Description</b>   | <b>RL enumerator</b> |
|------------------|--|----------------------|
| MS_Moving        | The object is currently moving.  | “exemplary”          |
| MS_StoppedMoving | The object movement was detected in the tracking history of the object.          | “exemplary”          |
| MS_Stationary    | Until now no object movement was detected in the tracking history of the object. | “exemplary”          |

### A.2.33 Number of valid lights

**Table A.108 — Signal: Number of valid lights**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid lights   |             |   |
| <b>Description</b> | The signal “Number of valid lights” (A.2.33) provides the current number of valid tuples [specifically the signal “Light type” (A.2.34) and the signal “Light status” (A.2.35) for a potentially moving object]. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

### A.2.34 Light type

**Table A.109 — Signal: Light type**

|             |            |
|-------------|------------|
| <b>Name</b> | Light type |
|-------------|------------|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | The signal “Light type” (A.2.34) provides a list of electric light classifications for a potentially moving object. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.110— Enumeration: Light type – Example enumerators**

| <b>Name</b>            | <b>Description</b>  | <b>RL enumerator</b> |
|------------------------|---|----------------------|
| LT_LeftTurnIndicator   | The light is the vehicle’s left turn indicator.   | “exemplary”          |
| LT_RightTurnIndicator  | The light is the vehicle’s right turn indicator.  | “exemplary”          |
| LT_HazardFlashLight    | The light is the vehicle’s hazard flash light.  | “exemplary”          |
| LT_LeftBrakeLight      | The light is the vehicle’s left brake light.  | “exemplary”          |
| LT_RightBrakeLight     | The light is the vehicle’s right brake light.   | “exemplary”          |
| LT_CentreBrakeLight    | The light is the vehicle’s centre brake light.  | “exemplary”          |
| LT_LeftOtherLight      | The light is the vehicle’s left light (no turn indicator, flash- or brake light).                                 | “exemplary”          |
| LT_RightOtherLight     | The light is the vehicle’s right light (no turn indicator, flash- or brake light).                                | “exemplary”          |
| LT_CentreOtherLight    | The light is a vehicle’s light which is not on the left or right side (no turn indicator, flash- or brake light). | “exemplary”          |
| LT_VehicleWarningLight | The light is the special vehicle’s warning light.   | “exemplary”          |
| LT_PoliceWarningLight  | The light is the police’s warning light.  | “exemplary”          |

**A.2.35 Light status****Table A.111— Signal: Light status**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Light status  |             |   |
| <b>Description</b> | <p>The signal “Light status” (A.2.35) provides the status of a potentially moving object’s electric light [see signal “Light type” (A.2.34)].</p> <p>Additional information:<br/>It is possible to have more than one active light signals (for example, low beam and blinker).</p> |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.112— Enumeration: Light status – Example enumerators**

| <b>Name</b>           | <b>Description</b>   | <b>RL enumerator</b> |
|-----------------------|--|----------------------|
| LS_Off                | The light status is off.   | “exemplary”          |
| LS_On                 | The light status is on.  | “exemplary”          |
| LS_LowFrequencyFlash  | The light status is cyclic flashing, for example, indicator.     | “exemplary”          |
| LS_HighFrequencyFlash | The light status indicates, for example, emergency full braking. | “exemplary”          |
| LS_Warning            | The light status indicates warning.                              | “exemplary”          |
| LS_Other              | The light status could not be determined.                        | “exemplary”          |

**A.2.36 Number of valid persons****Table A.113 — Signal: Number of valid persons**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid persons   |             |   |
| <b>Description</b> | The signal “Number of valid persons” (A.2.36) provides the current number of valid persons who are part of the potentially moving object. |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.2.37 Number of valid person type roles****Table A.114 — Signal: Number of valid person type roles**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid person type roles   |             |   |
| <b>Description</b> | The signal “Number of valid person type roles” (A.2.37) provides the current number of valid person type roles, which are classified for the potentially moving object. |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.2.38 Person type role****Table A.115 — Signal: Person type role**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Person type role  |             |   |
| <b>Description</b> | The signal “Person type role” (A.2.38) provides a classification of the person. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.116 — Enumeration: Person type role – Example enumerators**

| <b>Name</b>               | <b>Description</b>  | <b>RL enumerator</b> |
|---------------------------|---|----------------------|
| PTR_Policeman             | The person is a policeman.  | “exemplary”          |
| PTR_PolicemanInCar        | The person is a policeman, who gestures out of a police car.  | “exemplary”          |
| PTR_PolicemanOnMotorcycle | The person is a policeman, who gestures on a police motorcycle.                                       | “exemplary”          |
| PTR_RoadWorker            | The person is a road worker.  | “exemplary”          |
| PTR_PersonOfAuthority     | The person has special authority for defined circumstances, for example, firefighter, rescue service. | “exemplary”          |
| PTR_Person                | The person is an arbitrary person without any identified special characteristics/authorities.         | “exemplary”          |

**A.2.39 Person type role – confidence****Table A.117 — Signal: Person type role – confidence**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Person type role – confidence  |             |   |
| <b>Description</b> | The signal “Person type role – confidence” (A.2.39) provides the certainty for the corresponding signal “Person type role” (A.2.38). |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

## A.2.40 Number of valid person gestures

Table A.118 — Signal: Number of valid person gestures

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid person gestures   |             |   |
| <b>Description</b> | The signal “Number of valid person gestures” (A.2.40) provides the current number of valid person’s gestures. |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

## A.2.41 Person gesture indication type

Table A.119 — Signal: Person gesture indication type

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Person gesture indication type   |             |   |
| <b>Description</b> | The signal “Person gesture indication type” (A.2.41) provides the classification of the indication of a person’s gesture (for example, with hands). The gesture is made by a person defined in the signal “Person type role” (A.2.38). |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

Table A.120 — Enumeration: Person gesture indication type – Example enumerators

| <b>Name</b>                               | <b>Description</b>   | <b>RL enumerator</b> |
|---|--|----------------------|
| PGIT_RepetitiveHandMoving                 | The person repeats a gesture with its hand. This might be interpreted as trivial waving to say “hello” or to call attention to the person with this gesture because of any reason.   | “exemplary”          |
| PGIT_BendArmThumbBackwards                | The person bends one arm and stretches one thumb out of a fist. The forearm is moving back and forth. This gesture might be interpreted as a request for hitch hiking.   | “exemplary”          |
| PGIT_StretchedArmOpenHandToEgoVehicle     | The person stretched one arm in the vehicle’s direction with open hand. This gesture might be interpreted as a request to the vehicle to stop.   | “exemplary”          |
| PGIT_StretchedArmAlongPedestrianShoulders | The person stretches its arm to the left or right of the person’s body. This gesture might be interpreted as a proposed guiding or to proceed with the current driving. It can also be interpreted to follow a preceding vehicle.  | “exemplary”          |
| PGIT_StretchedArmToRight                  | The person stretches its arm to the right with respect to the ego vehicle coordinate system. This gesture might be interpreted as a proposal or request to drive to the right. It can also be interpreted as a hint to the driver to consider a referred other traffic participant or a specific traffic scene for the ego vehicle driving. It can also be interpreted to change to the right neighbouring lane. | “exemplary”          |



|                                   |   |             |
|-----------------------------------|---|-------------|
| PGIT_StretchedArmToLeft           | The person stretches its arm to the left with respect to the ego vehicle coordinate system. This gesture might be interpreted as a proposal or request to drive to the left. It can also be interpreted as a hint to the driver to consider a referred other traffic participant or a specific traffic scene for the ego vehicle driving. It can also be interpreted to change to the left neighbouring lane. | “exemplary” |
| PGIT_StretchedArmOpenHandToGround | The person stretches one or two arms to the ground. The open hands are directed to the ground. This might be interpreted to slow down or to stop the vehicle.   | “exemplary” |

#### A.2.42 Person gesture indication type – confidence

**Table A.121 — Signal: Person gesture direction type – confidence**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Person gesture direction type – confidence   |             |   |
| <b>Description</b> | The signal “Person gesture indication type – confidence” (A.2.42) provides the certainty for the corresponding signal “Person gesture indication type” (A.2.41). |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

#### A.2.43 Person gesture direction type

**Table A.122 — Signal: Person gesture direction type**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Person gesture direction type   |             |   |
| <b>Description</b> | The signal “Person gesture direction type” (A.2.43) provides the direction of an indicated request classification of a person’s gesture [see signal “Person gesture indication type” (A.2.41)] with respect to the ego vehicle. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.123 — Enumeration: Person gesture direction type – Example enumerators**

| <b>Name</b>           | <b>Description</b>  | <b>RL enumerator</b> |
|-----------------------|---|----------------------|
| PGDT_Straight         | The gesture is directed in driving direction of the ego vehicle. The direction of the person’s body does not matter.  | “exemplary”          |
| PGDT_Right            | The gesture is directed to the right with respect to the ego vehicle driving direction. The direction of the person’s body does not matter.                       | “exemplary”          |
| PGDT_Left             | The gesture is directed to the left with respect to the ego vehicle driving direction. The direction of the person’s body does not matter.                        | “exemplary”          |
| PGDT_StraightOpposite | The gesture is directed straight to the opposite direction with respect to the ego vehicle driving direction. The direction of the person’s body does not matter. | “exemplary”          |
| PGDT_RightBottom      | The gesture is directed to the bottom right with respect to the ego vehicle driving direction. The direction of the person’s body does not matter.                | “exemplary”          |

|                  |   |             |
|------------------|---|-------------|
| PGDT_LeftBottom  | The gesture is directed to the bottom left with respect to the ego vehicle driving direction. The direction of the person's body does not matter. | "exemplary" |
| PGDT_Unclear     | The gesture direction meaning is unclear.   | "exemplary" |
| PGDT_NoDirection | The persons gesture indicates no direction.   | "exemplary" |

#### A.2.44 Person gesture direction type – confidence

**Table A.124 — Signal: Person gesture direction type – confidence**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Person gesture direction type – confidence   |             |   |
| <b>Description</b> | The signal "Person gesture direction type – confidence" (A.2.44) provides the certainty for the corresponding signal "Person gesture direction type" (A.2.43). |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

#### A.2.45 Number of valid person's body part poses

**Table A.125 — Signal: Number of valid person's body part poses**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid person's body part poses  |             |   |
| <b>Description</b> | The signal "Number of valid person's body part poses" (A.2.45) provides the current number of valid tuples [specifically the signal "Person body part" (A.2.46) and the signal "Person body part orientation {yaw, pitch, roll}" (A.2.47)] for a potentially moving object. All body part poses are related to one person (for example, the object is a pedestrian, the object is a vehicle with a driver). |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

#### A.2.46 Person body part

**Table A.126 — Signal: Person body part**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Person body part  |             |   |
| <b>Description</b> | <p>The signal "Person body part" (A.2.46) indicates a body part. It is described by the signals "Person body part orientation {yaw, pitch, roll}" (A.2.47) and "Person body part origin {x, y, z}" (A.2.49).</p> <p>Additional information:<br/>The defined axis (normal vector) is always frontal or distal and defines the X-axis. To define an axis system the second axis is the corresponding distal or frontal axis (or cranial axis) and defines the Y-axis.</p> |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.127— Enumeration: Person body part – Example enumerators**

| <b>Name</b> | <b>Description</b>   | <b>RL enumerator</b> |
|-------------|--|----------------------|
| PBP_Head    | <p>The person's head (face director) is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).</p> <p>The normal vector to the face (eyes-nose plane) (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p> | "exemplary"          |

| Name              | Description   | RL enumerator |
|-------------------|---|---------------|
| PBP_UpperBody     | <p>The person's upper-body is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).</p> <p>The normal vector of the chest's front centre (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p>  | "exemplary"   |
| PBP_LeftHand      | <p>The person's left-hand is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).</p> <p>The normal vector of the hand palm (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p>  | "exemplary"   |
| PBP_RightHand     | <p>The person's right-hand is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).</p> <p>The normal vector of the hand palm (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p>   | "exemplary"   |
| PBP_LeftLowerArm  | <p>The person's left-lower arm is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).</p> <p>The long axis of the body extremities (orientation towards the end of the extremity, that means distal) (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p>  | "exemplary"   |
| PBP_RightLowerArm | <p>The person's right-lower arm is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).</p> <p>The long axis of the body extremities (orientation towards the end of the extremity, that means distal) (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p> | "exemplary"   |
| PBP_LeftUpperArm  | <p>The person's left-upper arm is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).</p> <p>The long axis of the body extremities (orientation towards the end of the extremity, that means distal) (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p>  | "exemplary"   |
| PBP_RightUpperArm | <p>The person's right-upper arm is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).</p> <p>The long axis of the body extremities (orientation towards the end of the extremity, that means distal) (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p> | "exemplary"   |
| PBP_LeftUpperLeg  | <p>The person's left-upper leg is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).</p> <p>The long axis of the body extremities (orientation towards the end of the extremity, that means distal) (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p>  | "exemplary"   |

| Name              | Description  | RL enumerator |
|-------------------|--|---------------|
| PBP_RightUpperLeg | The person's right-upper leg is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).<br>The long axis of the body extremities (orientation towards the end of the extremity, that means distal) (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)]. | "exemplary"   |
| PBP_LeftLowerLeg  | The person's left-lower leg is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).<br>The long axis of the body extremities (orientation towards the end of the extremity, that means distal) (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)].  | "exemplary"   |
| PBP_RightLowerLeg | The person's right-lower leg is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).<br>The long axis of the body extremities (orientation towards the end of the extremity, that means distal) (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)]. | "exemplary"   |
| PBP_LeftFoot      | The person's left-foot is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).<br>The foot sole direction towards the toe (frontal/distal) (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)].  | "exemplary"   |
| PBP_RightFoot     | The person's right-foot is defined by the corresponding signal "Person body part orientation {yaw, pitch, roll}" (A.2.47).<br>The foot sole direction towards the toe (frontal/distal) (see Figure A.7) is used to determine the pose relative to the ego-vehicle axis system [see signal "Vehicle coordinate system type – header" (A.1.20)].   | "exemplary"   |
| PBP_Chest         | The upper body may be divided in chest and abdominal area.   | "exemplary"   |
| PBP_AbdominalArea | The upper body may be divided in abdominal area and chest.   | "exemplary"   |
| PBP_LowerBody     | The legs may be covered by a skirt and could not be separated.   | "exemplary"   |
| PBP_UpperLegs     | The upper legs may be covered by a miniskirt and could not be separated.   | "exemplary"   |
| PBP_LowerLegs     | The legs may be covered by a skirt and the upper legs and lower legs may be divided in two parts (the person is wearing a skirt and is sitting on a bench), and the legs could not be separated.   | "exemplary"   |

**Key**

→ X-axis of the body part's coordinate system

→ Y-axis of the body part's coordinate system

**Figure A.7 — Example of person poses' coordinate systems**

#### A.2.47 Person body part orientation {yaw, pitch, roll}

**Table A.128 — Signal: Person body part orientation {yaw, pitch, roll}**

| Name | Person body part orientation {yaw, pitch, roll} |
|------|---|
|------|---|

|                    |  |             |                 |
|--------------------|--|-------------|-----------------|
| <b>Description</b> | <p>The signal “Person body part orientation {yaw, pitch, roll}” (A.2.47) provides the orientation of the body part for behaviour prediction (see Figure A.7).</p> <p>Additional information:</p> <p>The signal “Person body part orientation {yaw, pitch, roll}” (A.2.47) describes the angles of the pose’s orientation [defined by signal “Person body part” (A.2.46)] with respect to the ego-vehicle orientation, specifically the ego-vehicle axis system [see signal “Vehicle coordinate system type – header” (A.1.20)].</p> <p>The signal “Person body part orientation {yaw, pitch, roll} – error” (A.2.48) is given with {yaw, pitch, roll} angles, which have to be separated in a defined procedure.</p> <p>For example, the rotations are to be performed yaw first (around the Z-axis), pitch second (around the new Y-axis) and roll third (around the new X-axis) following the definition according to ISO 8855:2011.</p> |             |                 |
| <b>Value type</b>  | 3D vector real value   | <b>Unit</b> | (rad, rad, rad) |

#### A.2.48 Person body part orientation {yaw, pitch, roll} – error

**Table A.129 — Signal: Person body part orientation {yaw, pitch, roll} – error**

|                    |   |             |  |
|--------------------|---|-------------|--|
| <b>Name</b>        | Person body part orientation {yaw, pitch, roll} – error   |             |  |
| <b>Description</b> | The signal “Person body part orientation {yaw, pitch, roll} – error” (A.2.48) provides the error of the orientation of the body part [see signal “Person body part orientation {yaw, pitch, roll} – error” (A.2.48)]. |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal “Person body part orientation {yaw, pitch, roll} – error” (A.2.48). |

#### A.2.49 Person body part origin {x, y, z}

**Table A.130 — Signal: Person body part origin {x, y, z}**

|                    |   |             |           |
|--------------------|---|-------------|-----------|
| <b>Name</b>        | Person body part origin {x, y, z}   |             |           |
| <b>Description</b> | The signal “Person body part origin {x, y, z}” (A.2.49) provides the measured centre of the body part (midside, midwidth and midheight of the body part’s theoretical bounding box) of the person pose [see signal “Person body part orientation {yaw, pitch, roll}” (A.2.47)]. |             |           |
| <b>Value type</b>  | 3D vector real value  | <b>Unit</b> | (m, m, m) |

#### A.2.50 Person body part origin {x, y, z} – error

**Table A.131 — Signal: Person body part origin {x, y, z} – error**

|                    |   |             |  |
|--------------------|---|-------------|--|
| <b>Name</b>        | Person body part origin {x, y, z} – error   |             |  |
| <b>Description</b> | The signal “Person body part origin {x, y, z} – error” (A.2.50) provides the measured error of the signal “Person body part origin {x, y, z}” (A.2.49). |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal “Person body part origin {x, y, z}” (A.2.49). |

#### A.2.51 Eye state

**Table A.132 — Signal: Eye state**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Eye state   |             |   |
| <b>Description</b> | The signal “Eye state” (A.2.51) provides the classification of the person’s eyes. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.133 — Enumeration: Eye state – Example enumerators**

| <b>Name</b>      | <b>Description</b>  | <b>RL enumerator</b> |
|------------------|---|----------------------|
| ES_Open          | The person’s eyes are open.   | “exemplary”          |
| ES_Closed        | The person’s eyes are closed.   | “exemplary”          |
| ES_OpenAndClosed | One eye of the person is open and the second is closed.                                 | “exemplary”          |
| ES_NotVisible    | The person’s eyes are not visible. They may be hidden, for example, by a static object. | “exemplary”          |

**A.2.52 Eye state – confidence****Table A.134 — Signal: Eye state – confidence**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Eye state – confidence   |             |   |
| <b>Description</b> | The signal “Eye state – confidence” (A.2.52) provides the certainty for the corresponding signal “Eye state” (A.2.51). |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

**A.2.53 Viewing fixation time****Table A.135 — Signal: Viewing fixation time**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Viewing fixation time  |             |   |
| <b>Description</b> | The signal “Viewing fixation time” (A.2.53) provides the fixation duration of the person to the ego-vehicle. |             |   |
| <b>Value type</b>  | real value   | <b>Unit</b> | s |

**A.2.54 Viewing fixation time – error****Table A.136 — Signal: Viewing fixation time – error**

|                    |  |             |  |
|--------------------|--|-------------|--|
| <b>Name</b>        | Viewing fixation time – error  |             |  |
| <b>Description</b> | The signal “Viewing fixation time – error” (A.2.54) provides the error of the signal “Viewing fixation time” (A.2.53). |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Viewing fixation time” (A.2.53). |

**A.2.55 Viewing fixation direction****Table A.137 — Signal: Viewing fixation direction**

|                    |  |  |  |
|--------------------|--|--|--|
| <b>Name</b>        | Viewing fixation direction   |  |  |
| <b>Description</b> | The signal “Viewing fixation direction” (A.2.55) provides the fixation direction of the person to the ego-vehicle. |  |  |

|                   |             |             |   |
|-------------------|-------------|-------------|---|
| <b>Value type</b> | enumeration | <b>Unit</b> | 1 |
|-------------------|-------------|-------------|---|

**Table A.138— Enumeration: Viewing fixation direction - Example enumerators**

| <b>Name</b>                                 | <b>Description</b>   | <b>RL enumerator</b> |
|---|--|----------------------|
| VFD_EgoVehicleDirectFixation                | The person is fixating the ego-vehicle.  | “exemplary”          |
| VFD_EgoVehiclePeripheralFixation            | The person may have the ego-vehicle in the field of vision.  | “exemplary”          |
| VFD_EgoVehicleNoFixation                    | The person cannot have the ego-vehicle in the field of vision. The person’s viewing direction is, for example, in an opposite direction. | “exemplary”          |
| VFD_EgoVehicleFixationEstimationNotPossible | The direction of the person’s gaze cannot be determined.   | “exemplary”          |

**A.2.56 Viewing fixation direction - confidence****Table A.139 — Signal: Viewing fixation direction - confidence**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Viewing fixation direction - confidence  |             |   |
| <b>Description</b> | The signal “Viewing fixation direction - confidence” (A.2.56) provides the certainty for the corresponding signal “Viewing fixation direction” (A.2.55). |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

**A.2.57 Action type****Table A.140 — Signal: Action type**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Action type   |             |   |
| <b>Description</b> | The signal “Action type” (A.2.57) provides the classification of a person’s action (action with the whole body) [see also “Person gesture indication type” (A.2.41)]. For detailed information for moving persons like direction or velocity refer to “Velocity {x, y, z} - object level” (A.2.24). |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.141— Enumeration: Action type - Example enumerators**

| <b>Name</b>   | <b>Description</b>  | <b>RL enumerator</b> |
|---------------|---|----------------------|
| AT_Moving     | The person is moving, for example, walking, running, jumping, | “exemplary”          |
| AT_Bending    | The person is bending.  | “exemplary”          |
| AT_Lying      | The person is lying, for example, on the pavement.            | “exemplary”          |
| AT_Standstill | The person stands still and is not moving.                    | “exemplary”          |
| AT_Sitting    | The person is sitting, for example, on a bench.               | “exemplary”          |
| AT_Crawling   | The person is crawling on the floor.                          | “exemplary”          |

**A.2.58 Action type - confidence****Table A.142 — Signal: Action type - confidence**



|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Action type – confidence   |             |   |
| <b>Description</b> | The signal “Action type – confidence” (A.2.58) provides the certainty for the corresponding signal “Action type” (A.2.57). |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

#### A.2.59 Number of valid person accessories

**Table A.143 — Signal: Number of valid person accessories**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid person accessories   |             |   |
| <b>Description</b> | The signal “Number of valid person accessories” (A.2.59) provides the current number of valid accessories of the person. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

#### A.2.60 Person accessory classification type

**Table A.144 — Signal: Person accessory classification type**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Person accessory classification type   |             |   |
| <b>Description</b> | The signal “Person accessory classification type” (A.2.60) provides an accessory of the person. The accessory is not used for the motion of the potentially moving object [see signal “Potentially moving object classification type” (A.2.12)]. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.145 — Enumeration: Person accessory classification type – Example enumerators**

| <b>Name</b>            | <b>Description</b>   | <b>RL enumerator</b> |
|------------------------|--|----------------------|
| PACT_AssociatedObjects | The associated object entities to this potentially moving object are passive [see signal “Object grouping ID” (A.2.3)]. This potentially moving object is responsible for the motion of all other grouped object entities. | “exemplary”          |
| PACT_Suitcase          | The person has a suitcase as an accessory (if not recognised as separate potentially moving object).   | “exemplary”          |
| PACT_Umbrella          | The person has an umbrella as an accessory.  | “exemplary”          |
| PACT_Backpack          | The person has a backpack as an accessory.   | “exemplary”          |
| PACT_Trolley           | The person pulls a trolley as an accessory (if not recognised as separate potentially moving object).  | “exemplary”          |
| PACT_ShoppingCart      | The person pushes a shopping cart as an accessory (if not recognised as separate potentially moving object).   | “exemplary”          |
| PACT_HandCart          | The person pulls a hand cart as an accessory (if not recognised as separate potentially moving object).  | “exemplary”          |
| PACT_Ball              | The person plays with a ball (if not recognised as separate potentially moving object).  | “exemplary”          |
| PACT_Headphones        | The person is wearing headphones.  | “exemplary”          |
| PACT_FaceMask          | The person is wearing a face mask.   | “exemplary”          |
| PACT_SunGlasses        | The person is wearing sunglasses on its eyes.  | “exemplary”          |
| PACT_MealDrink         | The person is holding a meal or a drink in its hands.  | “exemplary”          |

| Name                         | Description                                   | RL enumerator |
|------------------------------|---|---------------|
| PACT_Helmet                  | The person is wearing a helmet.               | “exemplary”   |
| PACT_PolicemanSignallingDisk | The person uses a signalling disk to gesture. | “exemplary”   |
| PACT_Sign                    | The person uses a sign to gesture.            | “exemplary”   |

#### A.2.61 Person accessory classification type – confidence

**Table A.146 — Signal: Person accessory classification type – confidence**

| Name        | Person accessory classification type – confidence  |      |   |
|-------------|--|------|---|
| Description | The signal “Person accessory classification type – confidence” (A.2.61) provides the certainty for the corresponding signal “Person accessory classification type” (A.2.60). |      |   |
| Value type  | [0...100] real value   | Unit | % |

#### A.2.62 Accessory connection to person type

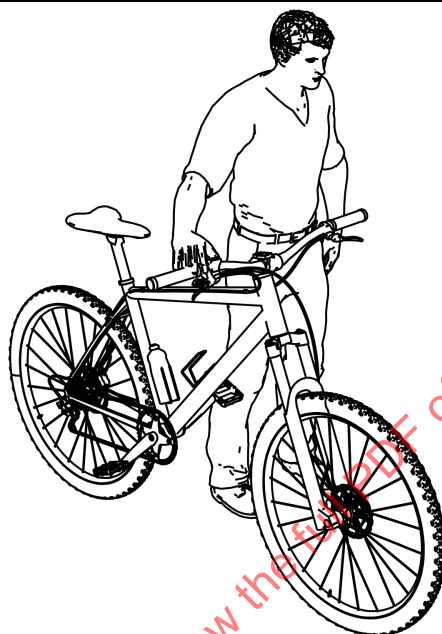
**Table A.147 — Signal: Accessory connection to person type**

| Name        | Accessory connection to person type   |      |   |
|-------------|---|------|---|
| Description | The signal “Accessory connection to person type” (A.2.62) provides the type of connection of the accessory to the person and the relative position with respect to the ego vehicle coordinate system. The signal “Person body part” (A.2.46) may provide the orientation of the referenced person poses [see signal “Person body part orientation {yaw, pitch, roll}” (A.2.47), “Person body part origin {x, y, z}” (A.2.49) and Figure A.8]. |      |   |
| Value type  | enumeration   | Unit | 1 |

**Table A.148 — Enumeration: Accessory connection to person type – Example enumerators**

| Name                        | Description  | RL enumerator |
|-----------------------------|--|---------------|
| ACTPT_HandLeftSideOfPerson  | The accessory is connected to the hand(s) of the person on the left side of the person with respect to the ego vehicle coordinate system.  | “exemplary”   |
| ACTPT_HandRightSideOfPerson | The accessory is connected to the hand(s) of the person on the right side of the person with respect to the ego vehicle coordinate system. | “exemplary”   |
| ACTPT_HandInFrontOfPerson   | The accessory is connected to the hand(s) in front of the person with respect to ego vehicle coordinate system X-axis direction.           | “exemplary”   |
| ACTPT_HandBehindOfPerson    | The accessory is connected to the hand(s) behind the person with respect to ego vehicle coordinate system X-axis direction.                | “exemplary”   |
| ACTPT_UpperBody             | The accessory is connected to the upper body of the person.  | “exemplary”   |
| ACTPT_FootLeftSideOfPerson  | The accessory is connected to a foot of the person on the left side of the person with respect to the ego vehicle coordinate system.       | “exemplary”   |
| ACTPT_FootRightSideOfPerson | The accessory is connected to a foot of the person on the right side of the person with respect to the ego vehicle coordinate system.      | “exemplary”   |

| Name                      | Description  | RL enumerator |
|---------------------------|--|---------------|
| ACTPT_FootInFrontOfPerson | The accessory is connected to a foot in front of the person with respect to the ego vehicle coordinate system. | "exemplary"   |
| ACTPT_FootBehindOfPerson  | The accessory is connected to a foot behind the person with respect to the ego vehicle coordinate system.      | "exemplary"   |



**Figure A.8** — Example of the signal "Accessory connection to person type" (A.2.62) with enumerator "ACTPT\_HandInFrontOfPerson"

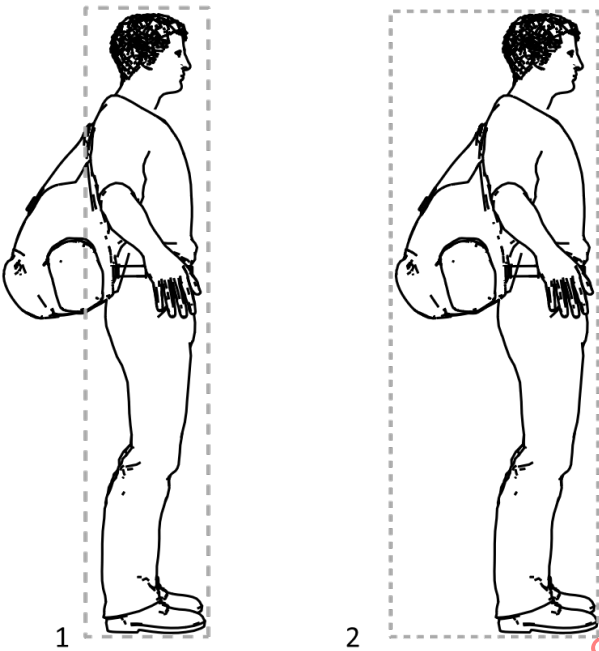
#### A.2.63 Included geometric structure of accessory

**Table A.149** — Signal: Included geometric structure of accessory

| Name        | Included geometric structure of accessory   |      |   |
|-------------|---|------|---|
| Description | The signal "Included geometric structure of accessory" (A.2.63) provides the information if the geometrical structure of the accessory is taken into account in the bounding box of the potentially moving object [see signal "Bounding box extent {length, width, height}" (A.2.20), "Included geometric structures" (A.2.23) and Figure A.9]. |      |   |
| Value type  | enumeration   | Unit | 1 |

**Table A.150** — Enumeration: Included geometric structure of accessory – Example enumerators

| Name                   | Description  | RL enumerator |
|------------------------|--|---------------|
| IGSOA_WithoutAccessory | The bounding box extent of the potentially moving object excludes accessory [see signal "Bounding box extent {length, width, height}" (A.2.20)]. | "exemplary"   |
| IGSOA_WithAccessory    | The bounding box extent of the potentially moving object includes accessory [see signal "Bounding box extent {length, width, height}" (A.2.20)]. | "exemplary"   |



Key

- 1 person's bounding box with signal "Included geometric structure of accessory" (A.2.63) := enumerator "IGSOA\_WithoutAccessory"
- 2 person's bounding box with signal "Included geometric structure of accessory" (A.2.63) := enumerator "IGSOA\_WithAccessory"

Figure A.9 — Example of the signal "Included geometric structure of accessory" (A.2.63)

A.2.64Object lane association

Table A.151 — Signal: Object lane association

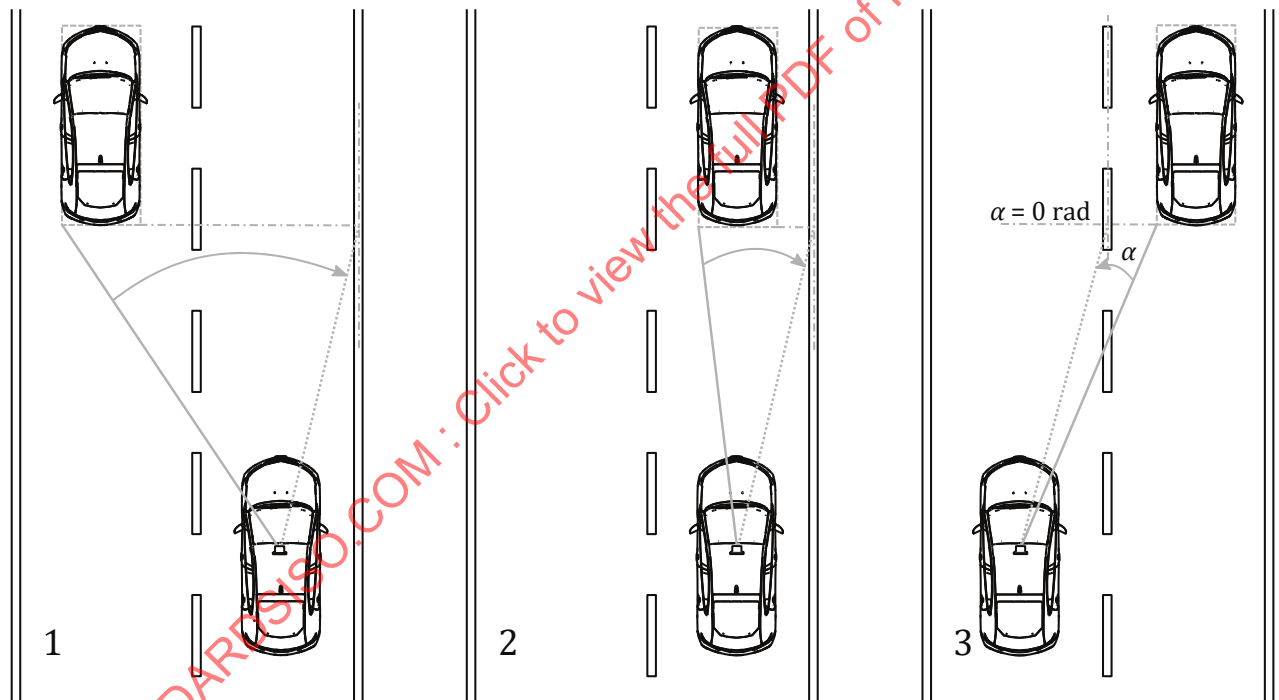
|             |   |      |   |
|-------------|---|------|---|
| Name        | Object lane association   |      |   |
| Description | The signal "Object lane association" (A.2.64) provides an association of the object to neighbouring lanes with respect to the ego-vehicle's lane. |      |   |
| Value type  | enumeration   | Unit | 1 |

Table A.152— Enumeration: Object lane association – Example enumerators

| Name              | Description  | RL enumerator |
|-------------------|--|---------------|
| OLA_EgoLane       | The object is in the ego lane.                                     | "exemplary"   |
| OLA_Left1Lane     | The object is in the first-left neighbouring lane.                 | "exemplary"   |
| OLA_Right1Lane    | The object is in the first-right neighbouring lane.                | "exemplary"   |
| OLA_EgoRight1Lane | The object is located between the ego and right neighbouring lane. | "exemplary"   |
| OLA_EgoLeft1Lane  | The object is located between the ego and left neighbouring lane.  | "exemplary"   |
| OLA_Left2Lane     | The object is in the second-left neighbouring lane.                | "exemplary"   |
| OLA_Right2Lane    | The object is in the second-right neighbouring lane.               | "exemplary"   |

**A.2.65 Angle between object edge and lane {left edge right lane, right edge left lane}****Table A.153 — Signal: Angle between object edge and lane {left edge right lane, right edge left lane}**

|                    |   |             |            |
|--------------------|---|-------------|------------|
| <b>Name</b>        | Angle between object edge and lane {left edge right lane, right edge left lane}   |             |            |
| <b>Description</b> | <p>{left edge right lane}: angle between the left edge of the object and the right centre of the ego-vehicle lane's road marking.</p> <p>{right edge left lane}: angle between the right edge of the object and the left centre of the ego-vehicle lane's road marking.</p> <p>Additional information:</p> <p>Signal, definition of 0 rad and positive angle is defined in Figure A.10 – key 3.</p> <p>The edge is always the nearest edge of the object's bounding box to the ego-vehicle.</p> <p>The centre of the road marking is the basis for the calculation.</p> |             |            |
| <b>Value type</b>  | 2D vector real value  | <b>Unit</b> | (rad, rad) |

**Key**

- 1 object is in the first-left neighbouring lane of the ego-vehicle
- 2 object is in the ego-vehicle lane
- 3 object is in the first-right neighbouring lane of the ego-vehicle
- $\alpha$  position angle {left edge right lane}

**Figure A.10 — Example for the signal: “Angle between object edge and lane {left edge right lane, right edge left lane}” (A.2.65) with left edge of the object and the right centre of the ego-vehicle lane's road marking**

**A.2.66 Angle between object edge and lane {left edge right lane, right edge left lane} – error****Table A.154 — Signal: Angle between object edge and lane {left edge right lane, right edge left lane} – error**

|                    |  |             |  |
|--------------------|--|-------------|--|
| <b>Name</b>        | Angle between object edge and lane {left edge right lane, right edge left lane} – error  |             |  |
| <b>Description</b> | The signal “Angle between object edge and lane {left edge right lane, right edge left lane} – error” (A.2.66) provides the error of the signal “Angle between object edge and lane {left edge right lane, right edge left lane}” (A.2.65). |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Angle between object edge and lane {left edge right lane, right edge left lane}” (A.2.65). |

**A.2.67 Percentage side lane {left, right}****Table A.155 — Signal: Percentage side lane {left, right}**

|                    |   |             |        |
|--------------------|---|-------------|--------|
| <b>Name</b>        | Percentage side lane {left, right}  |             |        |
| <b>Description</b> | <p>The signal “Percentage side lane {left, right}” (A.2.67) provides the {left, right} assignment of the object to specified lane in percent.</p> <p>Additional information:</p> <p>The signal “Percentage side lane {left, right}” (A.2.67) provides the percentage value of the object width in the corresponding {left, right} lane.</p> <p>The centre of the road marking is the basis for the calculation.</p> |             |        |
| <b>Value type</b>  | [0...100] 2D vector real value  | <b>Unit</b> | (%, %) |

**A.2.68 Angular position {azimuth}****Table A.156 — Signal: Angular position {azimuth}**

|                    |   |             |     |
|--------------------|---|-------------|-----|
| <b>Name</b>        | Angular position {azimuth}  |             |     |
| <b>Description</b> | <p>The signal “Angular position {azimuth}” (A.2.68) provides the lateral position of the bounding box centre (midside, midwidth, midheight) measured in the 2D image as {azimuth} angle in the sensor’s coordinate system.</p> <p>Additional information:</p> <p>It requires a calibrated sensor.</p> |             |     |
| <b>Value type</b>  | 1D vector real value  | <b>Unit</b> | rad |

**A.2.69 Angular velocity {azimuth}****Table A.157 — Signal: Angular velocity {azimuth}**

|                    |  |  |  |
|--------------------|--|--|--|
| <b>Name</b>        | Angular velocity {azimuth}   |  |  |
| <b>Description</b> | <p>The signal “Angular velocity {azimuth}” (A.2.69) provides the angular {azimuth} velocity of the bounding box centre (midside, midwidth, midheight) measured in the 2D image in the sensor’s coordinate system.</p> <p>Additional information:</p> <p>It requires a calibrated sensor.</p> |  |  |

|                   |                      |             |       |
|-------------------|----------------------|-------------|-------|
| <b>Value type</b> | 1D vector real value | <b>Unit</b> | rad/s |
|-------------------|----------------------|-------------|-------|

**A.2.70 Scale change – object level****Table A.158 — Signal: Scale change – object level**

|                    |  |             |     |
|--------------------|--|-------------|-----|
| <b>Name</b>        | Scale change – object level  |             |     |
| <b>Description</b> | The signal “Scale change – object level” (A.2.70) provides the change in scale of the object in sensor’s projection plane. It indicates whether the object gets closer to the ego-vehicle (possible collision) or if it is departing from the ego-vehicle. |             |     |
| <b>Value type</b>  | real value   | <b>Unit</b> | %/s |

**A.2.71 Entity radar cross section****Table A.159 — Signal: Entity radar cross section**

|                    |   |             |                   |
|--------------------|---|-------------|-------------------|
| <b>Name</b>        | Entity radar cross section  |             |                   |
| <b>Description</b> | The signal “Entity radar cross section” (A.2.71) provides the radar cross section (RCS) of the recognised object.<br><br>Additional information:<br>It is the representative RCS of the complete recognised object. |             |                   |
| <b>Value type</b>  | real value  | <b>Unit</b> | dB m <sup>2</sup> |

**A.2.72 Entity lidar reflectivity****Table A.160 — Signal: Entity lidar reflectivity**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Entity lidar reflectivity  |             |   |
| <b>Description</b> | The signal “Entity lidar reflectivity” (A.2.72) provides the lidar reflectivity of the recognised object, assuming homogeneous Lambertian reflectivity for the used laser wavelength.<br><br>Additional information:<br>It is the representative lidar reflectivity of the complete recognised object. |             |   |
| <b>Value type</b>  | real value   | <b>Unit</b> | % |

**A.2.73 Road type – object level****Table A.161 — Signal: Road type – object level**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Road type – object level   |             |   |
| <b>Description</b> | The signal “Road type – object level” (A.2.73) provides the relevant type of the road for the ego-vehicle. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.162 — Enumeration: Road type – object level – Example enumerators**

|             |  |                      |
|-------------|--|----------------------|
| <b>Name</b> | <b>Description</b>   | <b>RL enumerator</b> |
| RT_Motorway | The road characteristics meet motorway/highway/expressway/dual carriageway conditions. | “exemplary”          |

| Name            | Description   | RL enumerator |
|-----------------|---|---------------|
| RT_Interurban   | The road characteristics meet interurban road conditions. | "exemplary"   |
| RT_Rural        | The road characteristics meet rural conditions.           | "exemplary"   |
| RT_City         | The road characteristics meet city/urban conditions.      | "exemplary"   |
| RT_Suburb       | The road characteristics meet suburb conditions.          | "exemplary"   |
| RT_ParkingPlace | The road characteristics meet parking place conditions.   | "exemplary"   |
| RT_EnclosedPark | The road characteristics meet enclosed park conditions.   | "exemplary"   |
| RT_OffRoad      | The road characteristics meet off road conditions.        | "exemplary"   |

#### A.2.74 Road type – confidence

**Table A.163 — Signal: Road type – confidence**

| Name        | Road type – confidence  |      |   |
|-------------|---|------|---|
| Description | The signal "Road type – confidence" (A.2.74) provides the certainty for the corresponding signal "Road type – object level" (A.2.73). |      |   |
| Value type  | [0...100] real value  | Unit | % |

#### A.2.75 Number of valid road surface classifications

**Table A.164 — Signal: Number of valid road surface classifications**

| Name        | Number of valid road surface classifications  |      |   |
|-------------|---|------|---|
| Description | The signal "Number of valid road surface classifications" (A.2.75) provides the current number of valid tuples for road surface classifications. The tuples are defined by the signals "Road surface classification type – object level" (A.2.76) and "Road surface classification type – confidence" (A.2.77). |      |   |
| Value type  | [0...] integer value  | Unit | 1 |

#### A.2.76 Road surface classification type – object level

**Table A.165 — Signal: Road surface classification type – object level**

| Name        | Road surface classification type – object level   |      |   |
|-------------|---|------|---|
| Description | The signal "Road surface classification type – object level" (A.2.76) provides the relevant type of the road surface for the ego-vehicle. |      |   |
| Value type  | enumeration   | Unit | 1 |

**Table A.166 — Enumeration: Road surface classification type – object level – Example enumerators**

| Name           | Description  | RL enumerator |
|----------------|--|---------------|
| RSCT_Flat      | The road is a smooth-surface road.                 | "exemplary"   |
| RSCT_Concrete  | The road is a concrete road with a smooth surface. | "exemplary"   |
| RSCT_Asphalt   | The road is an asphalt road with a smooth surface. | "exemplary"   |
| RSCT_Bumpy     | The road is bumpy.                                 | "exemplary"   |
| RSCT_RomanRoad | The road is a cobblestone street.                  | "exemplary"   |



| Name            | Description                        | RL enumerator |
|-----------------|------------------------------------|---------------|
| RSCT_Gravel     | The road has a gravel surface.     | "exemplary"   |
| RSCT_Soil       | The road has a soil surface.       | "exemplary"   |
| RSCT_Sand       | The road has a sand surface.       | "exemplary"   |
| RSCT_OffRoad    | The road is an off-road road.      | "exemplary"   |
| RSCT_Vegetation | The road has a vegetation surface. | "exemplary"   |

#### A.2.77 Road surface classification type – confidence

**Table A.167 — Signal: Road surface classification type – confidence**

| Name        | Road surface classification type – confidence   |      |   |
|-------------|---|------|---|
| Description | The signal "Road surface classification type – confidence" (A.2.77) provides the certainty for the corresponding signal "Road surface classification type – object level" (A.2.76). |      |   |
| Value type  | [0...100] real value  | Unit | % |

#### A.2.78 Road surface roughness – object level

**Table A.168 — Signal: Road surface roughness – object level**

| Name        | Road surface roughness – object level   |      |      |
|-------------|---|------|------|
| Description | <p>The signal "Road surface roughness – object level" (A.2.78) provides the relevant roughness or unevenness of the road surface for the ego-vehicle.</p> <p>Additional information:<br/>The international roughness index (IRI) has a value range from 0 = mm/m (smooth surface which is equivalent to driving on a plate of glass) up to 20 mm/m (a very rough road).</p> |      |      |
| Value type  | [0...] real value   | Unit | mm/m |

#### A.2.79 Road surface texture – object level

**Table A.169 — Signal: Road surface texture – object level**

| Name        | Road surface texture – object level   |      |   |
|-------------|---|------|---|
| Description | The signal "Road surface texture – object level" (A.2.79) provides the surface texture or fine roughness [16] measures only wavelengths below 0,5 m, compared to the signal's value "Road surface roughness – object level" (A.2.78) (0,5 m – 100 m). |      |   |
| Value type  | real value  | Unit | m |

#### A.2.80 Number of valid road surface condition classifications

**Table A.170 — Signal: Number of valid road surface condition classifications**

| Name        | Number of valid road surface condition classifications  |  |  |
|-------------|---|--|--|
| Description | The signal "Number of valid road surface condition classifications" (A.2.80) provides the current number of valid tuples for road surface condition classifications. The tuples are defined by the signals "Road surface condition classification type – object level" (A.2.81) and "Road surface condition classification type – confidence" (A.2.82). |  |  |

|                   |                      |             |   |
|-------------------|----------------------|-------------|---|
| <b>Value type</b> | [0...] integer value | <b>Unit</b> | 1 |
|-------------------|----------------------|-------------|---|

**A.2.81 Road surface condition classification type – object level****Table A.171 — Signal: Road surface condition classification type – object level**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Road surface condition classification type – object level  |             |   |
| <b>Description</b> | The signal “Road surface condition classification type – object level” (A.2.81) provides the relevant condition of the road surface for the ego-vehicle. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.172 — Enumeration: Road surface condition classification type – object level – Example enumerators**

| <b>Name</b>  | <b>Description</b>                       | <b>RL enumerator</b> |
|--------------|--|----------------------|
| RSCCT_Dry    | The road surface is dry.                 | “exemplary”          |
| RSCCT_Wet    | The road surface is wet.                 | “exemplary”          |
| RSCCT_Water  | The road surface is covered with water.  | “exemplary”          |
| RSCCT_Snow   | The road surface is covered with snow.   | “exemplary”          |
| RSCCT_Ice    | The road surface is covered with ice.    | “exemplary”          |
| RSCCT_Leaves | The road surface is covered with leaves. | “exemplary”          |
| RSCCT_Dust   | The road surface is covered with dust.   | “exemplary”          |
| RSCCT_Dirt   | The road surface is covered with dirt.   | “exemplary”          |

**A.2.82 Road surface condition classification type – confidence****Table A.173 — Signal: Road surface condition classification type – confidence**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Road surface condition classification type – confidence   |             |   |
| <b>Description</b> | The signal “Road surface condition classification type – confidence” (A.2.82) provides the certainty for the corresponding signal “Road surface condition classification type – object level” (A.2.81). |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

**A.2.83 Number of valid road marking classifications****Table A.174 — Signal: Number of valid road marking classifications**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid road marking classifications   |             |   |
| <b>Description</b> | The signal “Number of valid road marking classifications” (A.2.83) provides the current number of valid tuples for road marking types. The tuples are defined by the signals “Road marking type” (A.2.84) and “Road marking type – confidence” (A.2.85). |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.2.84 Road marking type****Table A.175 — Signal: Road marking type**

|             |                   |  |  |
|-------------|-------------------|--|--|
| <b>Name</b> | Road marking type |  |  |
|-------------|-------------------|--|--|

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Description</b> | The signal “Road marking type” (A.2.84) indicates the type of the road markings. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.176— Enumeration: Road marking type – Example enumerators**

| <b>Name</b>                      | <b>Description</b>  | <b>RL enumerator</b> |
|----------------------------------|---|----------------------|
| RMT_Solid                        | The road marking is solid. It could also be a stop line.  | “exemplary”          |
| RMT_CentreLineDashedMarking      | The centre-line road marking is dashed.<br>See ISO 11270:2014, Annex B.   | “exemplary”          |
| RMT_EdgeLineDashedMarking        | The edge-line road marking is dashed.<br>See ISO 11270:2014, Annex B.   | “exemplary”          |
| RMT_Triangular                   | The road marking is a line of triangles.  | “exemplary”          |
| RMT_DoubleLineSolid              | The road marking has two lines and the most inner line (with respect to the ego-vehicle) is solid.  | “exemplary”          |
| RMT_CentreLineDoubleLineDashed   | The centre-line road marking has two lines and the most inner line (with respect to the ego-vehicle) is dashed.<br>See ISO 11270:2014, Annex B.           | “exemplary”          |
| RMT_EdgeLineDoubleLineDashed     | The edge-line road marking has two lines and the most inner line (with respect to the ego-vehicle) is dashed.<br>See ISO 11270:2014, Annex B.             | “exemplary”          |
| RMT_MultipleLineSolid            | The road marking has more than two lines and the most inner line (with respect to the ego-vehicle) is solid.  | “exemplary”          |
| RMT_CentreLineMultipleLineDashed | The centre-line road marking has more than two lines and the most inner line (with respect to the ego-vehicle) is dashed.<br>See ISO 11270:2014, Annex B. | “exemplary”          |
| RMT_EdgeLineMultipleLineDashed   | The edge-line road marking has more than two lines and the most inner line (with respect to the ego-vehicle) is dashed.<br>See ISO 11270:2014, Annex B.   | “exemplary”          |
| RMT_BottsDotsCatsEyes            | The road marking consists of Botts' dots or cats' eyes.   | “exemplary”          |
| RMT_AttentionMarker              | The road marking is an attention marker, for example, US, China and Japan.  | “exemplary”          |
| RMT_Hatched                      | The road marking is hatched/chevron.  | “exemplary”          |
| RMT_Box                          | The road marking of a junction.   | “exemplary”          |
| RMT_ColouredArea                 | The road marking is a coloured area.  | “exemplary”          |
| RMT_Arrow                        | The road marking is an arrow.   | “exemplary”          |
| RMT_ZebraCrossing                | The road marking is a zebra crossing/continental/ladder.  | “exemplary”          |
| RMT_GenericSymbol                | The road marking is a generic symbol or icon.   | “exemplary”          |
| RMT_TrafficSignOnLane            | The road marking is a traffic sign.   | “exemplary”          |

|                        |   |             |
|------------------------|---|-------------|
| RMT_GenericLine        | The road marking is a generic line.                           | “exemplary” |
| RMT_ParkingArea        | The road marking is a parking area.                           | “exemplary” |
| RMT_TShapeMarkingBegin | The road marking is a parking T-shape beginning parking line. | “exemplary” |
| RMT_TShapeMarkingEnd   | The road marking is a parking T-shape ending parking line.    | “exemplary” |
| RMT_IShapeMarkingBegin | The road marking is a parking I-shape beginning parking line. | “exemplary” |
| RMT_IShapeMarkingEnd   | The road marking is a parking I-shape ending parking line.    | “exemplary” |
| RMT_LShapeMarkingBegin | The road marking is a parking L-shape beginning parking line. | “exemplary” |
| RMT_LShapeMarkingEnd   | The road marking is a parking L-shape ending parking line.    | “exemplary” |
| RMT_Nets               | The road marking is a net, specifically a non-stopping area.  | “exemplary” |

#### A.2.85 Road marking type – confidence

**Table A.177 — Signal: Road marking type – confidence**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Road marking type – confidence  |             |   |
| <b>Description</b> | The signal “Road marking type – confidence” (A.2.85) provides the confidence of the classified signal “Road marking type” (A.2.84) information. |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

#### A.2.86 Road object lane association

**Table A.178 — Signal: Road object lane association**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Road object lane association  |             |   |
| <b>Description</b> | The signal “Road object lane association” (A.2.86) provides the association of a road marking or a road boundary to a lane with respect to the ego-vehicle lane, that means the ego lane. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.179 — Enumeration: Road object lane association – Example enumerators**

| <b>Name</b>        | <b>Description</b>  | <b>RL enumerator</b> |
|--------------------|---|----------------------|
| ROLA_EgoLane       | The road marking is on the ego lane.  | “exemplary”          |
| ROLA_EgoLeft1Lane  | The road boundary separates the ego lane from 1 <sup>st</sup> left neighbouring lane.<br>The road marking is associated to the ego lane and the 1 <sup>st</sup> left neighbouring lane.   | “exemplary”          |
| ROLA_EgoRight1Lane | The road boundary separates the ego lane from 1 <sup>st</sup> right neighbouring lane.<br>The road marking is associated to the ego lane and the 1 <sup>st</sup> right neighbouring lane. | “exemplary”          |
| ROLA_Left1Lane     | The road marking is on the 1 <sup>st</sup> left neighbouring lane.  | “exemplary”          |

|                       |   |             |
|-----------------------|---|-------------|
| ROLA_Right1Lane       | The road marking is on the 1 <sup>st</sup> right neighbouring lane.   | “exemplary” |
| ROLA_Left1Left2Lane   | The road boundary separates the 1 <sup>st</sup> left lane from the 2 <sup>nd</sup> left neighbouring lane.<br>The road marking is associated to the 1 <sup>st</sup> left and the 2 <sup>nd</sup> left neighbouring lanes.     | “exemplary” |
| ROLA_Right1Right2Lane | The road boundary separates the 1 <sup>st</sup> right lane from the 2 <sup>nd</sup> right neighbouring lane.<br>The road marking is associated to the 1 <sup>st</sup> right and the 2 <sup>nd</sup> right neighbouring lanes. | “exemplary” |
| ROLA_Left2Lane        | The road marking is on the 2 <sup>nd</sup> left neighbouring lane.  | “exemplary” |
| ROLA_Right2Lane       | The road marking is on the 2 <sup>nd</sup> right neighbouring lane.   | “exemplary” |
| ROLA_LeftRoadEdge     | The road boundary limits at the outer edge of the leftmost lane.  | “exemplary” |
| ROLA_RightRoadEdge    | The road boundary limits at the outer edge of the rightmost lane.   | “exemplary” |

### A.2.87 Road object lane association – confidence

**Table A.180 — Signal: Road object lane association – confidence**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Road object lane association – confidence  |             |   |
| <b>Description</b> | The signal “Road object lane association – confidence” (A.2.87) provides the confidence of the signal’s “Road object lane association” (A.2.86) information. |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

### A.2.88 Arrow orientation {yaw}

**Table A.181 — Signal: Arrow orientation {yaw}**

|                    |   |             |     |
|--------------------|---|-------------|-----|
| <b>Name</b>        | Arrow orientation {yaw}   |             |     |
| <b>Description</b> | The signal “Arrow orientation {yaw}” (A.2.88) provides the orientation of the arrow symbol on the ground plane with respect to the driving direction of the ego-vehicle.<br><br>Additional information:<br>The signal “Arrow orientation {yaw}” (A.2.88) := 0 rad, if the signal “Road marking type” (A.2.84) does not have the enumerator “RMT_Arrow”. |             |     |
| <b>Value type</b>  | 1D vector real value  | <b>Unit</b> | rad |

### A.2.89 Arrow direction

**Table A.182 — Signal: Arrow direction**

|                    |   |  |  |
|--------------------|---|--|--|
| <b>Name</b>        | Arrow direction   |  |  |
| <b>Description</b> | The signal “Arrow direction” (A.2.89) provides the estimated direction of the displayed arrow on the lane.<br><br>Additional information:<br>The enumerator “AD_Foreward”, which essentially means 0 rad, points in the direction of driving straight ahead on the associated lane.<br>The signal “Arrow direction” (A.2.89) := “AD_Foreward”, if the signal “Road marking type” (A.2.84) does not have the enumerator “RMT_Arrow”. |  |  |

|                   |             |             |   |
|-------------------|-------------|-------------|---|
| <b>Value type</b> | enumeration | <b>Unit</b> | 1 |
|-------------------|-------------|-------------|---|

**Table A.183 — Enumeration: Arrow direction – Example enumerators**

| Name                 | Description   | RL enumerator |
|----------------------|---|---------------|
| AD_Foreward          | The arrow has an estimated direction of 0 rad (0 °).  | “exemplary”   |
| AD_Left              | The arrow has an estimated direction of $+\frac{\pi}{2}$ rad (+90 °).                       | “exemplary”   |
| AD_Right             | The arrow has an estimated direction of $-\frac{\pi}{2}$ rad (−90 °).                       | “exemplary”   |
| AD_StraightLeft      | The arrow is straight left and has an estimated direction of $+\frac{\pi}{2}$ rad (+90 °).  | “exemplary”   |
| AD_StraightRight     | The arrow is straight right and has an estimated direction of $-\frac{\pi}{2}$ rad (−90 °). | “exemplary”   |
| AD_TurningPointLeft  | The arrow has an estimated direction of $+\pi$ rad (+180 °), that means a U-turn left.      | “exemplary”   |
| AD_TurningPointRight | The arrow has an estimated direction of $-\pi$ rad (−180 °), that means a U-turn right.     | “exemplary”   |
| AD_45DegLeft         | The arrow has an estimated direction of $+\frac{\pi}{4}$ rad (+45 °).                       | “exemplary”   |
| AD_45DegRight        | The arrow has an estimated direction of $-\frac{\pi}{4}$ rad (−45 °).                       | “exemplary”   |
| AD_NoArrow           | No arrow is present.  | “exemplary”   |

**A.2.90 Number of valid sign classifications****Table A.184 — Signal: Number of valid sign classifications**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid sign classifications   |             |   |
| <b>Description</b> | The signal “Number of valid sign classifications” (A.2.90) provides the current number of valid tuples for sign classifications. The tuples are defined by the signals “Sign classification type” (A.2.91) and “Sign classification type – confidence” (A.2.92). |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.2.91 Sign classification type****Table A.185 — Signal: Sign classification type**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Sign classification type  |             |   |
| <b>Description</b> | The signal “Sign classification type” (A.2.91) provides the type of the sign for a main sign or a sign as a road marking. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.186 — Enumeration: Sign classification type – Example enumerators**

| Name               | Description                     | RL enumerator |
|--------------------|---------------------------------|---------------|
| SCT_StopSign       | The sign is a stop sign.        | “exemplary”   |
| SCT_YieldSign      | The sign is a yield sign.       | “exemplary”   |
| SCT_SpeedLimitSign | The sign is a speed limit sign. | “exemplary”   |

| Name                | Description   | RL enumerator |
|---------------------|---|---------------|
| SCT_NoMainSign      | The sign is no main sign: it is only one or multiple supplementary signs.           | "exemplary"   |
| SCT_GreenArrowSign  | The sign is a green arrow sign at traffic lights.                                   | "exemplary"   |
| SCT_HeightLimitSign | The sign is a height limit sign.  | "exemplary"   |
| SCT_WidthLimitSign  | The sign is a width limit sign.   | "exemplary"   |
| SCT_MessageSign     | The sign is a message sign.   | "exemplary"   |
| SCT_EmptySign       | The sign may be a changeable traffic sign without displaying a traffic sign symbol. | "exemplary"   |

#### A.2.92 Sign classification type – confidence

**Table A.187 — Signal: Sign classification type – confidence**

| Name        | Sign classification type – confidence  |      |   |
|-------------|--|------|---|
| Description | The signal "Sign classification type – confidence" (A.2.92) provides the confidence for the signal's "Sign classification type" (A.2.91) classification information, for example, the degree of certainty that the sign is of the reported type, for example, stop sign. |      |   |
| Value type  | [0...100] real value   | Unit | % |

#### A.2.93 Sign value

**Table A.188 — Signal: Sign value**

| Name        | Sign value  |      |   |
|-------------|---|------|---|
| Description | The signal "Sign value" (A.2.93) provides the depicted numerical value on the sign. |      |   |
| Value type  | real value  | Unit | The unit is defined by the enumerator of the signal "Sign value unit" (A.2.94). |

#### A.2.94 Sign value unit

**Table A.189 — Signal: Sign value unit**

| Name        | Sign value unit  |      |   |
|-------------|--|------|---|
| Description | The signal "Sign value unit" (A.2.94) provides the unit which the numerical value of the signal "Sign value" (A.2.93) is referring to. |      |   |
| Value type  | enumeration  | Unit | 1 |

**Table A.190 — Enumeration: Sign value unit – Example enumerators**

| Name                 | Description  | RL enumerator |
|----------------------|--|---------------|
| SVU_KilometrePerHour | The unit for the sign value is kilometre per hour. | "exemplary"   |
| SVU_MilePerHour      | The unit for the sign value is mile per hour.      | "exemplary"   |
| SVU_Metre            | The unit for the sign value is metre.              | "exemplary"   |
| SVU_Kilometre        | The unit for the sign value is kilometre.          | "exemplary"   |
| SVU_Feet             | The unit for the sign value is feet.               | "exemplary"   |
| SVU_Mile             | The unit for the sign value is mile.               | "exemplary"   |

| Name           | Description   | RL enumerator |
|----------------|---|---------------|
| SVU_MetricTon  | The unit for the sign value is metric ton.  | “exemplary”   |
| SVU_LongTon    | The unit for the sign value is long ton.  | “exemplary”   |
| SVU_ShortTon   | The unit for the sign value is short ton.   | “exemplary”   |
| SVU_Minute     | The unit for the sign value is minute.  | “exemplary”   |
| SVU_Hour       | The unit for the sign value is hour.  | “exemplary”   |
| SVU_Day        | The unit for the sign value is the day of the month.  | “exemplary”   |
| SVU_Weekday    | The unit for the sign value is the weekday.<br>The numeric value of the weekday is defined as:<br>- Sunday [Sign value (A.2.93) := 0]<br>- ...<br>- Saturday [Sign value (A.2.93) := 6] | “exemplary”   |
| SVU_Percentage | The unit for the sign value is percentage.  | “exemplary”   |

## A.2.95 Sign text

Table A.191 — Signal: Sign text

|             |  |      |   |
|-------------|--|------|---|
| Name        | Sign text  |      |   |
| Description | The signal “Sign text” (A.2.95) provides the text message on the sign. The message may be multiline. |      |   |
| Value type  | string   | Unit | 1 |

## A.2.96 Sign text font

Table A.192 — Signal: Sign text font

|             |  |      |   |
|-------------|--|------|---|
| Name        | Sign text font   |      |   |
| Description | The signal “Sign text font” (A.2.96) describes the font style of the text on the sign. |      |   |
| Value type  | enumeration  | Unit | 1 |

Table A.193 — Enumeration: Sign text font – Example enumerators

| Name           | Description  | RL enumerator |
|----------------|--|---------------|
| STF_Regular    | The font style of the text on the sign is regular.         | “exemplary”   |
| STF_Bold       | The font style of the text on the sign is bold.            | “exemplary”   |
| STF_Italic     | The font style of the text on the sign is italic.          | “exemplary”   |
| STF_BoldItalic | The font style of the text on the sign is bold and italic. | “exemplary”   |

## A.2.97 Sign text – confidence

Table A.194 — Signal: Sign text – confidence

|             |   |  |  |
|-------------|---|--|--|
| Name        | Sign text – confidence  |  |  |
| Description | The signal “Sign text – confidence” (A.2.97) provides the confidence for the text signal “Sign text” (A.2.95) and the text font signal “Sign text font” (A.2.96). |  |  |



|                   |                      |             |   |
|-------------------|----------------------|-------------|---|
| <b>Value type</b> | [0...100] real value | <b>Unit</b> | % |
|-------------------|----------------------|-------------|---|

## A.2.98 Number of valid entity states

**Table A.195 — Signal: Number of valid entity states**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid entity states  |             |   |
| <b>Description</b> | The signal “Number of valid entity states” (A.2.98) provides the current number of valid states [see signal “Entity state” (A.2.99)] of an entity, for example, a sign, a traffic sign board region. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

## A.2.99 Entity state

**Table A.196 — Signal: Entity state**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Entity state  |             |   |
| <b>Description</b> | The signal “Entity state” (A.2.99) describes the state of an entity of signs or traffic sign boards. It describes, for example, whether or not the message of the entity is variable/changeable, static or if that information is not available. It describes also, if, for example, the traffic sign is out of service, because the symbol is crossed out. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.197 — Enumeration: Entity state – Example enumerators**

| <b>Name</b>              | <b>Description</b>   | <b>RL enumerator</b> |
|--------------------------|--|----------------------|
| ES_Static                | The message sign is a static (non-variable) message sign.          | “exemplary”          |
| ES_Variable              | The message sign is a variable message sign.                       | “exemplary”          |
| ES_VariableNonAnimated   | The message sign is a variable message sign without animation.     | “exemplary”          |
| ES_VariableAnimated      | The message sign is a variable message sign with animation.        | “exemplary”          |
| ES_SwitchedOff           | The message sign is a variable message sign which is switched off. | “exemplary”          |
| ES_FullOutOfService      | The message sign is full out of service.                           | “exemplary”          |
| ES_PartiallyOutOfService | Part of the message sign is out of service.                        | “exemplary”          |
| ES_OutOfView             | The message sign has rotated.                                      | “exemplary”          |
| ES_PartiallyOccluded     | The message sign is partly occluded.                               | “exemplary”          |

## A.2.100 Colour value – object level

**Table A.198 — Signal: Colour value – object level**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Colour value – object level  |             |   |
| <b>Description</b> | <p>The signal “Colour value – object level” (A.2.100) provides the colour value(s). The number of values depends on the signal “Colour model type” (A.1.14).</p> <p>Additional information:<br/>For example, enumerator “CMT_RGB” indicates 3 colour values, specifically first value red, second value green, third value blue.</p> |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

## A.2.101 Colour tone – confidence – object level

Table A.199 — Signal: Colour tone – confidence – object level

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Colour tone – confidence – object level   |             |   |
| <b>Description</b> | The signal “Colour tone – confidence – object level” (A.2.101) provides the confidence of the colour tone. The colour tone is defined by the signal “Colour model type” (A.1.14) and the list of signals “Colour value – object level” (A.2.100). |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

## A.2.102 Number of valid connections

Table A.200 — Signal: Number of valid connections

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid connections  |             |   |
| <b>Description</b> | The signal “Number of valid connections” (A.2.102) provides the current number of valid tuples for connections. The tuples are defined by the signal “Connection type” (A.2.103) and unique signal’s “Connection grouping ID” (A.2.104) value for each connection. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

## A.2.103 Connection type

Table A.201 — Signal: Connection type

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Connection type   |             |   |
| <b>Description</b> | The signal “Connection type” (A.2.103) provides the type of connection of at least two road marking’s polylines or polynomials. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

Table A.202 — Enumeration: Connection type – Example enumerators

| <b>Name</b>        | <b>Description</b>   | <b>RL enumerator</b> |
|--------------------|--|----------------------|
| CT_Merge           | The connection of road markings is a merge (from the point of view of the ego-vehicle) of road markings. | “exemplary”          |
| CT_Split           | The connection of road markings is a split (from the point of view of the ego-vehicle) of road markings. | “exemplary”          |
| CT_Interconnection | The connection of road markings is an interconnection of road markings.                                  | “exemplary”          |
| CT_Extension       | The connection of road markings is an extension of two road markings.                                    | “exemplary”          |

## A.2.104 Connection grouping ID

Table A.203 — Signal: Connection grouping ID

|             |                        |
|-------------|------------------------|
| <b>Name</b> | Connection grouping ID |
|-------------|------------------------|

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Description</b> | <p>The signal “Connection grouping ID” (A.2.104) provides a unique grouping ID linking all connected road boundaries together.</p> <p>Additional information:<br/>The signal “Connection grouping ID” (A.2.104) shall be unique in each cycle of the sensor cluster.</p> |             |   |
| <b>Value type</b>  | [...] integer value  | <b>Unit</b> | 1 |

**A.2.105 Vertex point {x, y, z}****Table A.204 — Signal: Vertex point {x, y, z}**

|                    |  |             |           |
|--------------------|--|-------------|-----------|
| <b>Name</b>        | Vertex point {x, y, z}   |             |           |
| <b>Description</b> | The signal “Vertex point {x, y, z}” (A.2.105) provides the measured longitudinal, lateral and vertical distance of the vertex point. |             |           |
| <b>Value type</b>  | 3D vector real value   | <b>Unit</b> | (m, m, m) |

**A.2.106 Vertex point {x, y, z} – error****Table A.205 — Signal: Vertex point {x, y, z} – error**

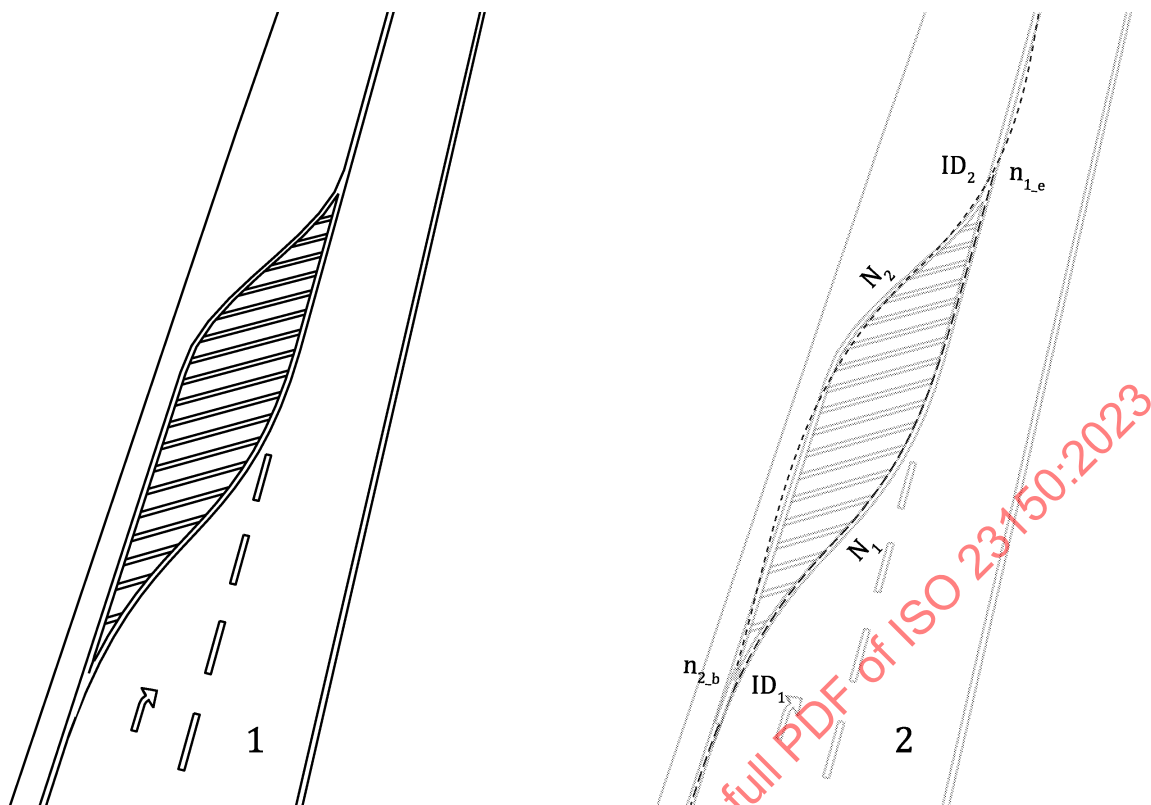
|                    |   |             |  |
|--------------------|---|-------------|--|
| <b>Name</b>        | Vertex point {x, y, z} – error  |             |  |
| <b>Description</b> | The signal “Vertex point {x, y, z} – error” (A.2.106) provides the measured error of the signal “Vertex point {x, y, z}” (A.2.105). |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal “Vertex point {x, y, z}” (A.2.105). |

**A.2.107 Vertex point – confidence {x, y, z}****Table A.206 — Signal: Vertex point – confidence {x, y, z}**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Vertex point – confidence {x, y, z}   |             |   |
| <b>Description</b> | The signal “Vertex point – confidence {x, y, z}” (A.2.107) provides the confidence of the measured longitudinal, lateral and vertical distance of the vertex point. |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

**A.2.108 Number of valid polynomials****Table A.207 — Signal: Number of valid polynomials**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid polynomials  |             |   |
| <b>Description</b> | The signal “Number of valid polynomials” (A.2.108) provides the current number of valid polynomials to describe the shape (see Figure A.11). |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |



- Key**
- 1 real-world scene
  - 2 polynomial contour representation with 2 polynomials and 2 road marking connections
  - $N_x$   $x^{\text{th}}$  polynomial
  - $n_{x,b}$  range begin of the  $x^{\text{th}}$  polynomial [see “Polynomial range  $x$  {begin, end}” (A.2.113)]
  - $n_{x,e}$  range end of the  $x^{\text{th}}$  polynomial [see “Polynomial range  $x$  {begin, end}” (A.2.113)]
  - $ID_x$  road marking connection - signal “Connection grouping ID” (A.2.104) :=  $x$

**Figure A.11 — Example for a polynomial contour representation**

**A.2.109 Polynomial coefficient  $y \{c_0, c_1, c_2, c_3\}$**

**Table A.208 — Signal: Polynomial coefficient  $y \{c_0, c_1, c_2, c_3\}$**

| Name        | Polynomial coefficient $y \{c_0, c_1, c_2, c_3\}$   |      |                                |
|-------------|---|------|--------------------------------|
| Description | <p>The signal “Polynomial coefficient <math>y \{c_0, c_1, c_2, c_3\}</math>” (A.2.109) provides the calculated coefficients for the Y-axis value of the polynomial line.</p> <p>Calculation method:</p> $y = c_0 + c_1 \cdot x + c_2 \cdot x^2 + c_3 \cdot x^3$ <p>where:</p> <ul style="list-style-type: none"><li><math>c_0</math>: is the coefficient for degree 0,</li><li><math>c_1</math>: is the coefficient for degree 1,</li><li><math>c_2</math>: is the coefficient for degree 2,</li><li><math>c_3</math>: is the coefficient for degree 3 and</li><li><math>x</math>: is the value of the X-axis in m [see signal “Polynomial range <math>x</math> {begin, end}” (A.2.113)].</li></ul> |      |                                |
| Value type  | 4D vector real value  | Unit | (m, 1, 1/m, 1/m <sup>2</sup> ) |

**A.2.110 Polynomial coefficient z {c<sub>0</sub>, c<sub>1</sub>, c<sub>2</sub>, c<sub>3</sub>}****Table A.209 — Signal: Polynomial coefficient z {c<sub>0</sub>, c<sub>1</sub>, c<sub>2</sub>, c<sub>3</sub>}**

|                    |   |             |                                |
|--------------------|---|-------------|--------------------------------|
| <b>Name</b>        | Polynomial coefficient z {c <sub>0</sub> , c <sub>1</sub> , c <sub>2</sub> , c <sub>3</sub> }   |             |                                |
| <b>Description</b> | <p>The signal “Polynomial coefficient z {c<sub>0</sub>, c<sub>1</sub>, c<sub>2</sub>, c<sub>3</sub>}” (A.2.110) provides the calculated coefficients for the Z-axis value of the polynomial line.</p> <p>Calculation method:</p> $z = c_0 + c_1 \cdot x + c_2 \cdot x^2 + c_3 \cdot x^3$ <p>where:</p> <p>c<sub>0</sub>: is the coefficient for degree 0,<br/> c<sub>1</sub>: is the coefficient for degree 1,<br/> c<sub>2</sub>: is the coefficient for degree 2,<br/> c<sub>3</sub>: is the coefficient for degree 3 and<br/> x: is the value of the X-axis in m [see signal “Polynomial range x {begin, end}” (A.2.113)].</p> |             |                                |
| <b>Value type</b>  | 4D vector real value  | <b>Unit</b> | (m, 1, 1/m, 1/m <sup>2</sup> ) |

**A.2.111 Polynomial y – error****Table A.210 — Signal: Polynomial y – error**

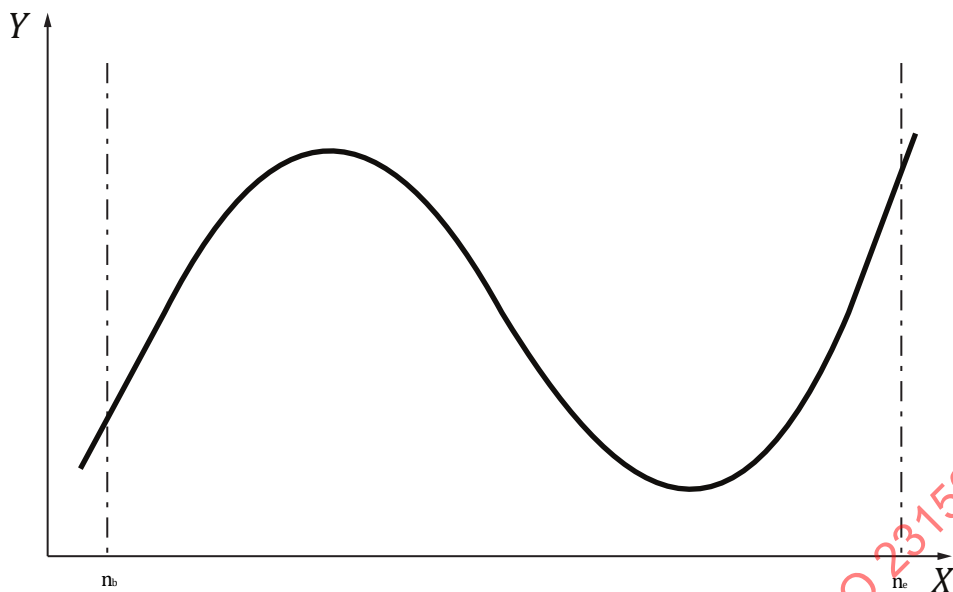
|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Polynomial y – error   |             |   |
| <b>Description</b> | The signal “Polynomial y – error” (A.2.111) provides the error of the Y-axis value of the polynomial line. The unit and the error definition for the signal “Polynomial y – error” (A.2.111) are defined during the system design phase. |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Polynomial coefficient y {c <sub>0</sub> , c <sub>1</sub> , c <sub>2</sub> , c <sub>3</sub> }” (A.2.109). |

**A.2.112 Polynomial z – error****Table A.211 — Signal: Polynomial z – error**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Polynomial z – error   |             |   |
| <b>Description</b> | The signal “Polynomial z – error” (A.2.112) provides the error of the Z-axis value of the polynomial line. The unit and the error definition for the signal “Polynomial z – error” (A.2.112) are defined during the system design phase. |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Polynomial coefficient z {c <sub>0</sub> , c <sub>1</sub> , c <sub>2</sub> , c <sub>3</sub> }” (A.2.110). |

**A.2.113 Polynomial range x {begin, end}****Table A.212 — Signal: Polynomial range x {begin, end}**

|                    |  |             |        |
|--------------------|--|-------------|--------|
| <b>Name</b>        | Polynomial range x {begin, end}  |             |        |
| <b>Description</b> | The signal “Polynomial range x {begin, end}” (A.2.113) provides the valid range of the polynomials [x <sub>Begin</sub> , x <sub>End</sub> ] (see Figure A.12). |             |        |
| <b>Value type</b>  | 2D vector real value   | <b>Unit</b> | (m, m) |



**Key**  
X-axis and Y-axis: vehicle axis system  
 $n_b, n_e$  {begin} and {end} of signal “Polynomial range x {begin, end}” (A.2.113)

**Figure A.12 — Example for the signal “Polynomial range x {begin, end}” (A.2.113) for signal “Polynomial coefficient y {c<sub>0</sub>, c<sub>1</sub>, c<sub>2</sub>, c<sub>3</sub>}” (A.2.109)**

**A.2.114 Extent {width, height} – polynomial**

**Table A.213 — Signal: Extent {width, height} – polynomial**

|                    |  |             |        |
|--------------------|--|-------------|--------|
| <b>Name</b>        | Extent {width, height} – polynomial  |             |        |
| <b>Description</b> | The signal “Extent {width, height} – polynomial” (A.2.114) provides the measured width and height at the X-axis value $(x_{\text{Begin}} + x_{\text{End}})/2$ [see “Polynomial range x {begin, end}” (A.2.113)]. The width information is relative to the polynomial line which defines the centre of the width range. The height of the road object is added to the polynomial z value. |             |        |
| <b>Value type</b>  | 2D vector real value   | <b>Unit</b> | (m, m) |

**A.2.115 Extent {width, height} – polynomial – error**

**Table A.214 — Signal: Extent {width, height} – polynomial – error**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Extent {width, height} – polynomial – error  |             |   |
| <b>Description</b> | The signal “Extent {width, height} – polynomial – error” (A.2.115) provides the error of the signal “Extent {width, height} – polynomial” (A.2.114). |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Extent {width, height} – polynomial” (A.2.114). |

**A.2.116 Extent {width, height} – polynomial – confidence**

**Table A.215 — Signal: Extent {width, height} – polynomial – confidence**

|             |  |
|-------------|--|
| <b>Name</b> | Extent {width, height} – polynomial – confidence |
|-------------|--|

|                    |   |             |        |
|--------------------|---|-------------|--------|
| <b>Description</b> | The signal “Extent {width, height} – polynomial – confidence” (A.2.116) provides the confidence of the width and height for the polynomial segment. |             |        |
| <b>Value type</b>  | 2D vector [0...100] real value  | <b>Unit</b> | (%, %) |

### A.2.117 Number of valid data ranges

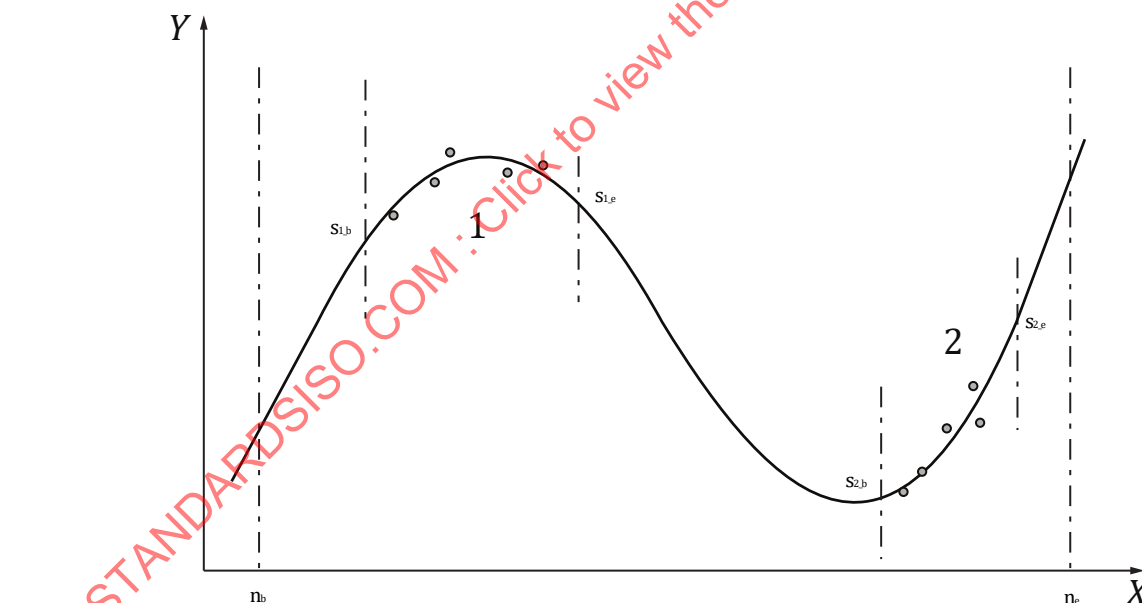
**Table A.216 — Signal: Number of valid data ranges**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid data ranges  |             |   |
| <b>Description</b> | The signal “Number of valid data ranges” (A.2.117) provides the current number of valid <i>X</i> -axis data ranges for the polynomial lines. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

### A.2.118 Supported data range x {begin, end}

**Table A.217 — Signal: Supported data range x {begin, end}**

|                    |  |             |        |
|--------------------|--|-------------|--------|
| <b>Name</b>        | Supported data range x {begin, end}  |             |        |
| <b>Description</b> | The signal “Supported data range x {begin, end}” (A.2.118) provides the supported range of the polynomial [ <i>X</i> <sub>Begin</sub> , <i>X</i> <sub>End</sub> ] which is covered with measured points (see Figure A.13). |             |        |
| <b>Value type</b>  | 2D vector real value   | <b>Unit</b> | (m, m) |



#### Key

- X*-axis and *Y*-axis: vehicle axis system
- 1 supported data range 1 with measured points
- 2 supported data range 2 with measured points
- $n_b$ ,  $n_e$  {begin} and {end} of signal “Polynomial range x {begin, end}” (A.2.113)
- $s_{x,b}$ ,  $s_{x,e}$  {begin} and {end} of the  $x^{\text{th}}$  signal “Supported data range x {begin, end}” (A.2.118)

**Figure A.13 — Example for the signal “Supported data range x {begin, end}” (A.2.118) for signal “Polynomial coefficient y { $c_0$ ,  $c_1$ ,  $c_2$ ,  $c_3$ }” (A.2.109)**

## A.2.119 Supported axis

Table A.218 — Signal: Supported axis

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Supported axis  |             |   |
| <b>Description</b> | The signal “Supported axis” (A.2.119) provides the information of the polynomial's supported axis for the signal “Supported data range x {begin, end}” (A.2.118). |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

Table A.219 — Enumeration: Supported axis

| <b>Name</b> | <b>Description</b>  | <b>RL enumerator</b>   |
|-------------|---|--|
| SA_Y        | The signal “Supported data range x {begin, end}” (A.2.118) corresponds to Y-axis polynomial line of signal “Polynomial coefficient y {c0, c1, c2, c3}” (A.2.109).   | Mandatory  |
| SA_Z        | The signal “Supported data range x {begin, end}” (A.2.118) corresponds to Z-axis polynomial line of signal “Polynomial coefficient z {c0, c1, c2, c3}” (A.2.110).   | Optional   |
| SA_YAndZ    | The signal “Supported data range x {begin, end}” (A.2.118) corresponds to both polynomial lines of signal “Polynomial coefficient y {c0, c1, c2, c3}” (A.2.109) and signal “Polynomial coefficient z {c0, c1, c2, c3}” (A.2.110). | Conditional (B.2.2)<br>* The enumeration has the enumerator “SA_Z” |

## A.2.120 Polyline interpolation method

Table A.220 — Signal: Polyline interpolation method

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Polyline interpolation method  |             |   |
| <b>Description</b> | <p>The signal “Polyline interpolation method” (A.2.120) provides the method for interpolating between the points of a polyline.</p> <p>Additional information:<br/>Interpolation method may be different for Y-axis and Z-axis values.</p> |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

Table A.221 — Enumeration: Polyline interpolation method - Example enumerators

| <b>Name</b> | <b>Description</b>  | <b>RL enumerator</b> |
|-------------|---|----------------------|
| PIM_Linear  | The polyline is interpolated with a linear interpolation between two sequential points. | “exemplary”          |
| PIM_Spline  | The polyline is interpolated with a spline interpolation between two sequential points. | “exemplary”          |
| PIM_Cubic   | The polyline is interpolated with a cubic interpolation between two sequential points.  | “exemplary”          |

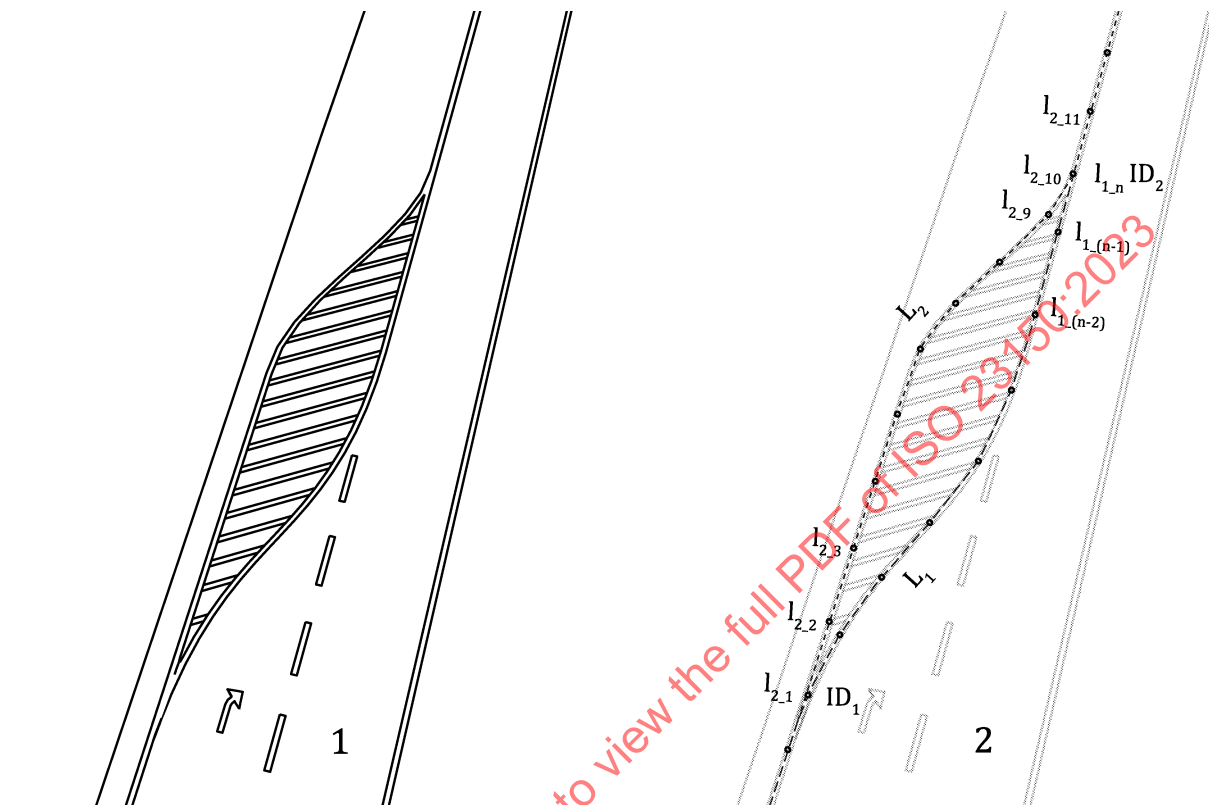
## A.2.121 Number of valid polylines

Table A.222 — Signal: Number of valid polylines

|             |                           |
|-------------|---------------------------|
| <b>Name</b> | Number of valid polylines |
|-------------|---------------------------|



|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Description</b> | The signal “Number of valid polylines” (A.2.121) provides the current number of valid polylines to describe a shape with multiple polylines (see Figure A.14). |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**Key**

- 1 real-world scene
- 2 polyline contour representation with 2 polylines (multiple vertices) and 2 road marking connections
- $L_x$   $x^{\text{th}}$  polyline
- $l_{x,y}$   $y^{\text{th}}$  vertex point [signal “Vertex point {x, y, z}” (A.2.105)] of the  $x^{\text{th}}$  polyline
- $ID_x$  road marking connection - signal “Connection grouping ID” (A.2.104) := x

**Figure A.14 — Example for a polyline contour representation****A.2.122 Number of valid vertices****Table A.223 — Signal: Number of valid vertices**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid vertices   |             |   |
| <b>Description</b> | <p>The signal “Number of valid vertices” (A.2.122) provides the current number of valid vertices of a polyline. A polyline describes a part of the shape border with one continuous segmented line (see Figure A.14) with at least 2 vertex points.</p> <p>Additional information:</p> <p>The signal “Extent {width, height} - vertex” (A.2.123) has attributes of the vertex at the point of the signal “Vertex point {x, y, z}” (A.2.105). The error and confidence signals apply to this point.</p> |             |   |
| <b>Value type</b>  | [2...] integer value   | <b>Unit</b> | 1 |

## A.2.123 Extent {width, height} – vertex

Table A.224 — Signal: Extent {width, height} – vertex

|                    |  |             |        |
|--------------------|--|-------------|--------|
| <b>Name</b>        | Extent {width, height} – vertex  |             |        |
| <b>Description</b> | <p>The signal “Extent {width, height} – vertex” (A.2.123) provides the measured width and height of the vertex at the vertex point of the signal “Vertex point {x, y, z}” (A.2.105). The width information is relative to the vertex which defines the centre of the width range. The height of the road object is added to the vertex point z value.</p> <p>Additional information:<br/>The signal applies to the point of the signal “Vertex point {x, y, z}” (A.2.105).</p> |             |        |
| <b>Value type</b>  | 2D vector real value   | <b>Unit</b> | (m, m) |

## A.2.124 Extent {width, height} – vertex – error

Table A.225 — Signal: Extent {width, height} – vertex – error

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Extent {width, height} – vertex – error  |             |   |
| <b>Description</b> | <p>The signal “Extent {width, height} – vertex – error” (A.2.124) provides the error of the measured width and height [see signal “Extent {width, height} – vertex” (A.2.123)] at the vertex point [see signal “Vertex point {x, y, z}” (A.2.105)].</p> <p>Additional information:<br/>The signal applies to the point of the signal “Vertex point {x, y, z}” (A.2.105).</p> |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Extent {width, height} – vertex” (A.2.123). |

## A.2.125 Extent {width, height} – vertex – confidence

Table A.226 — Signal: Extent {width, height} – vertex – confidence

|                    |  |             |        |
|--------------------|--|-------------|--------|
| <b>Name</b>        | Extent {width, height} – vertex – confidence   |             |        |
| <b>Description</b> | <p>The signal “Extent {width, height} – vertex – confidence” (A.2.125) provides the confidence of the measured width and height of the vertex [see signal “Extent {width, height} – vertex” (A.2.123)].</p> <p>Additional information:<br/>The signal applies to the point of the signal “Vertex point {x, y, z}” (A.2.105).</p> |             |        |
| <b>Value type</b>  | 2D vector [0...100] real value   | <b>Unit</b> | (%, %) |

## A.2.126 Number of valid road boundary classifications

Table A.227 — Signal: Number of valid road boundary classifications

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid road boundary classifications  |             |   |
| <b>Description</b> | <p>The signal “Number of valid road boundary classifications” (A.2.126) provides the current number of valid tuples for road boundary types. The tuples are defined by the signals “Road boundary type” (A.2.127), “Road boundary type – confidence” (A.2.128) and additional signals.</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.2.127 Road boundary type****Table A.228 — Signal: Road boundary type**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Road boundary type   |             |   |
| <b>Description</b> | The signal “Road boundary type” (A.2.127) indicates the type of the road boundary. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.229 — Enumeration: Road boundary type – Example enumerators**

| <b>Name</b>            | <b>Description</b>  | <b>RL enumerator</b> |
|------------------------|---|----------------------|
| RBT_GuardRail          | The road boundary is a guard rail.  | “exemplary”          |
| RBT_Fence              | The road boundary is a fence.   | “exemplary”          |
| RBT_Wall               | The road boundary is a wall, for example, brick wall, building.                       | “exemplary”          |
| RBT_Barrier            | The road boundary is a barrier.   | “exemplary”          |
| RBT_TensionCableSystem | The road boundary is a tension cable system.  | “exemplary”          |
| RBT_RoadEdge           | The road boundary is a road edge, for example, grass, vegetation, sand, gravel, soil. | “exemplary”          |
| RBT_Curb               | The road boundary is a curb stone.  | “exemplary”          |

**A.2.128 Road boundary type – confidence****Table A.230 — Signal: Road boundary type – confidence**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Road boundary type – confidence   |             |   |
| <b>Description</b> | The signal “Road boundary type – confidence” (A.2.128) provides the confidence of the classified signal “Road boundary type” (A.2.127). |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

**A.2.129 Number of valid general landmark classifications****Table A.231 — Signal: Number of valid general landmark classifications**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid general landmark classifications  |             |   |
| <b>Description</b> | The signal “Number of valid general landmark classifications” (A.2.129) provides the current number of valid tuples for general landmark classifications. The tuples are defined by the signals “General landmark classification type” (A.2.130) and “General landmark classification type – confidence” (A.2.131). |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.2.130 General landmark classification type****Table A.232 — Signal: General landmark classification type**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | General landmark classification type   |             |   |
| <b>Description</b> | The signal “General landmark classification type” (A.2.130) provides the classification of the general landmark. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.233 — Enumeration: General landmark classification type – Example enumerators**

| Name                      | Description                                      | RL enumerator |
|---------------------------|--|---------------|
| GLCT_Bridge               | The general landmark is a bridge.                | “exemplary”   |
| GLCT_Beacon               | The general landmark is a beacon.                | “exemplary”   |
| GLCT_Cone                 | The general landmark is a cone.                  | “exemplary”   |
| GLCT_Barrel               | The general landmark is a barrel.                | “exemplary”   |
| GLCT_GuidePost            | The general landmark is a guide post.            | “exemplary”   |
| GLCT_LampPost             | The general landmark is a lamp post.             | “exemplary”   |
| GLCT_VerticalStructure    | The general landmark is a vertical structure.    | “exemplary”   |
| GLCT_OverheadObject       | The general landmark is an overhead object.      | “exemplary”   |
| GLCT_RectangularStructure | The general landmark is a rectangular structure. | “exemplary”   |
| GLCT_Tunnel               | The general landmark is a tunnel.                | “exemplary”   |
| GLCT_Reflector            | The general landmark is a reflector.             | “exemplary”   |
| GLCT_OverridableObject    | The general landmark is an overridable object.   | “exemplary”   |

**A.2.131 General landmark classification type – confidence****Table A.234 — Signal: General landmark classification type – confidence**

| Name        | General landmark classification type – confidence   |      |   |
|-------------|---|------|---|
| Description | The signal “General landmark classification type – confidence” (A.2.131) provides the confidence for the corresponding signal “General landmark classification type” (A.2.130). |      |   |
| Value type  | [0...100] real value  | Unit | % |

**A.2.132 Sign geometry****Table A.235 — Signal: Sign geometry**

| Name        | Sign geometry   |      |   |
|-------------|---|------|---|
| Description | The signal “Sign geometry” (A.2.132) provides the shape geometry of the sign. |      |   |
| Value type  | enumeration   | Unit | 1 |

**Table A.236 — Enumeration: Sign geometry – Example enumerators**

| Name            | Description   | RL enumerator |
|-----------------|---|---------------|
| SG_Circle       | The sign has a circle shape as sign geometry.                               | “exemplary”   |
| SG_TriangleTop  | The sign has a triangle shape with tip pointing downwards as sign geometry. | “exemplary”   |
| SG_TriangleDown | The sign has a triangle shape with tip pointing upwards as sign geometry.   | “exemplary”   |
| SG_Square       | The sign has a square shape as sign geometry.                               | “exemplary”   |
| SG_Pole         | The sign has a pole shape as sign geometry.                                 | “exemplary”   |
| SG_Rectangle    | The sign has a rectangle shape as sign geometry.                            | “exemplary”   |
| SG_Diamond      | The sign has a diamond shape as sign geometry.                              | “exemplary”   |

| Name                 | Description  | RL enumerator |
|----------------------|--|---------------|
| SG_ArrowLeft         | The sign has an arrow left, five edge shape as sign geometry.  | "exemplary"   |
| SG_ArrowRight        | The sign has an arrow right, five edge shape as sign geometry. | "exemplary"   |
| SG_Octagon           | The sign has an octagon shape as sign geometry.                | "exemplary"   |
| SG_SaintAndrewsCross | The sign has a cross shape as sign geometry.                   | "exemplary"   |
| SG_GateShape         | The sign has a gate shape as sign geometry.                    | "exemplary"   |

### A.2.133 Number of valid lane relevance classifications

**Table A.237 — Signal: Number of valid lane relevance classifications**

| Name        | Number of valid lane relevance classifications  |      |   |
|-------------|---|------|---|
| Description | The signal "Number of valid lane relevance classifications" (A.2.133) provides the current number of valid tuples for lane relevance classifications. The tuples are defined by the signals "Lane relevance classification type" (A.2.134) and "Lane relevance classification type - confidence" (A.2.135). |      |   |
| Value type  | [0...] integer value  | Unit | 1 |

### A.2.134 Lane relevance classification type

**Table A.238 — Signal: Lane relevance classification type**

| Name        | Lane relevance classification type  |      |   |
|-------------|---|------|---|
| Description | The signal "Lane relevance classification type" (A.2.134) provides the information regarding if the sign is relevant for the ego-vehicle's lane, the nearest lane to the ego-vehicle or other relevant lanes. |      |   |
| Value type  | enumeration   | Unit | 1 |

**Table A.239 — Enumeration: Lane relevance classification type - Example enumerators**

| Name                  | Description   | RL enumerator |
|-----------------------|---|---------------|
| LRCT_EgoLane          | The entity is relevant on track of ego-vehicle.                                       | "exemplary"   |
| LRCT_Left1Lane        | The entity is relevant for the next lane to the ego-vehicle on the left side.         | "exemplary"   |
| LRCT_Right1Lane       | The entity is relevant for the next lane to the ego-vehicle on the right side.        | "exemplary"   |
| LRCT_Left2Lane        | The entity is relevant for the second next lane to the ego-vehicle on the left side.  | "exemplary"   |
| LRCT_Right2Lane       | The entity is relevant for the second next lane to the ego-vehicle on the right side. | "exemplary"   |
| LRCT_EgoAndLeft1Lane  | The entity is relevant on track of ego-vehicle and the next left lane.                | "exemplary"   |
| LRCT_EgoAndRight1Lane | The entity is relevant on track of ego-vehicle and the next right lane.               | "exemplary"   |
| LRCT_LeftmostLane     | The entity is relevant for the leftmost lane.   | "exemplary"   |
| LRCT_RightmostLane    | The entity is relevant for the rightmost lane.  | "exemplary"   |

| Name           | Description   | RL enumerator |
|----------------|---|---------------|
| LRCT_AllLanes  | The entity is relevant for all lanes, which essentially means the lane(s) to the right- and left side and on track. | “exemplary”   |
| LRCT_OtherLane | The entity is relevant for another far lane.  | “exemplary”   |

#### A.2.135 Lane relevance classification type – confidence

**Table A.240 — Signal: Lane relevance classification type – confidence**

| Name        | Lane relevance classification type – confidence   |      |   |
|-------------|---|------|---|
| Description | The signal “Lane relevance classification type – confidence” (A.2.135) provides the confidence for the signal’s “Lane relevance classification type” (A.2.134) classification information, for example, the degree of certainty that the lane relevance is of the reported type, for example, relevant for all lanes. |      |   |
| Value type  | [0...100] real value  | Unit | % |

#### A.2.136 Number of valid traffic supplementary signs

**Table A.241 — Signal: Number of valid traffic supplementary signs**

| Name        | Number of valid traffic supplementary signs   |      |   |
|-------------|---|------|---|
| Description | The signal “Number of valid traffic supplementary signs” (A.2.136) provides the current number of valid supplementary sign entities.<br><br>Additional information:<br>The entries in the array may or may not be sorted. |      |   |
| Value type  | [0...] integer value  | Unit | 1 |

#### A.2.137 Number of valid supplementary sign classifications

**Table A.242 — Signal: Number of valid supplementary sign classifications**

| Name        | Number of valid supplementary sign classifications  |      |   |
|-------------|---|------|---|
| Description | The signal “Number of valid supplementary sign classifications” (A.2.137) provides the current number of valid tuples for supplementary sign classifications. The tuples are defined by the signals “Supplementary sign classification type” (A.2.138) and “Supplementary sign classification type – confidence” (A.2.139). |      |   |
| Value type  | [0...] integer value  | Unit | 1 |

#### A.2.138 Supplementary sign classification type

**Table A.243 — Signal: Supplementary sign classification type**

| Name        | Supplementary sign classification type   |      |   |
|-------------|--|------|---|
| Description | The signal “Supplementary sign classification type” (A.2.138) provides the type of the sign. |      |   |
| Value type  | enumeration  | Unit | 1 |

**Table A.244 — Enumeration: Supplementary sign classification type – Example enumerators**

| Name                       | Description   | RL enumerator |
|----------------------------|---|---------------|
| SSCT_ValidInformationBegin | The sign displays a begin of valid zone.  | “exemplary”   |
| SSCT_ValidInformationEnd   | The sign displays an end of valid zone.   | “exemplary”   |
| SSCT_Frost                 | The sign displays a frost sign.   | “exemplary”   |
| SSCT_WetRoad               | The sign displays a wet road sign.  | “exemplary”   |
| SSCT_ValidInDistance       | The sign displays a distance information and the main sign becomes valid in the defined distance. | “exemplary”   |
| SSCT_ValidForDistance      | The sign displays a distance information and the main sign is valid for the defined distance.     | “exemplary”   |
| SSCT_Limitation            | The sign displays a limitation information.   | “exemplary”   |

**A.2.139 Supplementary sign classification type – confidence****Table A.245 — Signal: Supplementary sign classification type – confidence**

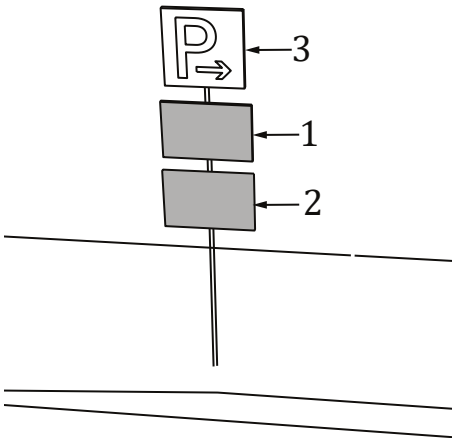
| Name        | Supplementary sign classification type – confidence   |      |   |
|-------------|---|------|---|
| Description | The signal “Supplementary sign classification type – confidence” (A.2.139) provides the confidence for the signal’s “Supplementary sign classification type” (A.2.138) classification information, for example, the degree of certainty that the supplementary traffic sign is of the reported type, for example, limitation information. |      |   |
| Value type  | [0...100] real value  | Unit | % |

**A.2.140 Relative position****Table A.246 — Signal: Relative position**

| Name        | Relative position   |      |   |
|-------------|---|------|---|
| Description | The signal “Relative position” (A.2.140) provides the relative position of the supplemental sign with respect to its main sign (see Figure A.15). |      |   |
| Value type  | enumeration   | Unit | 1 |

**Table A.247 — Enumeration: Relative position – Example enumerators**

| Name     | Description                                       | RL enumerator |
|----------|---|---------------|
| RP_Above | The supplementary sign is above the main sign.    | “exemplary”   |
| RP_Below | The supplementary sign is below the main sign.    | “exemplary”   |
| RP_Left  | The supplementary sign is left of the main sign.  | “exemplary”   |
| RP_Right | The supplementary sign is right of the main sign. | “exemplary”   |



Key

- 1 supplementary sign entity with signal “Relative position” (A.2.140) := enumerator “RP\_Below” and signal “Relative position order” (A.2.141) := 1
- 2 supplementary sign entity with signal “Relative position” (A.2.140) := enumerator “RP\_Below” and signal “Relative position order” (A.2.141) := 2
- 3 main sign entity of the supplementary signs

Figure A.15 — Example for a traffic sign – a main sign with two supplementary signs

A.2.141 Relative position order

Table A.248 — Signal: Relative position order

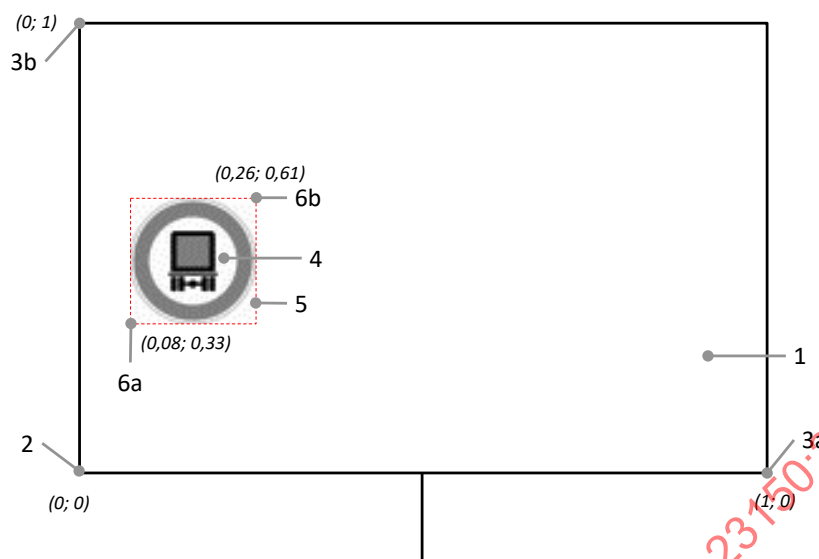
|             |   |      |   |
|-------------|---|------|---|
| Name        | Relative position order   |      |   |
| Description | The signal “Relative position order” (A.2.141) provides the relative order of a supplementary sign with respect to the main sign position. The value 1 defines the nearest position of the supplementary sign to the main sign (see Figure A.15). |      |   |
| Value type  | integer value   | Unit | 1 |

A.2.142 Position uv\_origin {x, y, z}

Table A.249 — Signal: Position uv\_origin {x, y, z}

|             |   |      |           |
|-------------|---|------|-----------|
| Name        | Position uv_origin {x, y, z}  |      |           |
| Description | <p>The signal “Position uv_origin {x, y, z}” (A.2.142) provides the position of the reference corner of a sign board which defines the origin of the single sign board affine object coordinate system. It is defined in the ego-vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)] (see Figure A.16).</p> <p>Additional information:</p> <p>Even though not always all corners are visible, the origin “Position uv_origin {x, y, z}” (A.2.142) shall be always the left lower corner. The direction of the affine coordinate system is defined by the end positions “Position u_end {x, y, z}” (A.2.144) in negative Y direction of the ego-vehicle coordinate system and “Position v_end {x, y, z}” (A.2.146) in the positive Z direction of the ego-vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)].</p> |      |           |
| Value type  | 3D vector real value  | Unit | (m, m, m) |





### Key

- 1 traffic sign board
- 2 origin of the affine coordinate system of the traffic sign board – signal “Position uv\_origin {x, y, z}” (A.2.142) → affine coordinates (0; 0)
- 3a abscissa of the affine coordinate system of the traffic sign board – signal “Position u\_end {x, y, z}” (A.2.144) → affine coordinates (1; 0)
- 3b ordinate of the affine coordinate system of the traffic sign board – signal “Position v\_end {x, y, z}” (A.2.146) → affine coordinates (0; 1)
- 4 referenced traffic sign as traffic main sign – signal “Object ID reference – object level” (A.2.179)
- 5 bounding box of the referenced traffic sign as traffic main sign (key 4)
- 6a {u<sub>Begin</sub>, v<sub>Begin</sub>} of the bounding box (key 5) – signal with affine coordinates
- 6b {u<sub>End</sub>, v<sub>End</sub>} of the bounding box (key 5) – “Bounding box {u<sub>Begin</sub>, u<sub>End</sub>, v<sub>Begin</sub>, v<sub>End</sub>}” (A.2.173) := (0,08; 0,26; 0,33; 0,61)

**Figure A.16 — Affine object coordinate system of a traffic sign board**

### A.2.143 Position uv\_origin {x, y, z} – error

**Table A.250 — Signal: Position uv\_origin {x, y, z} – error**

| Name        | Position uv_origin {x, y, z} – error   |      |  |
|-------------|--|------|--|
| Description | The signal “Position uv_origin {x, y, z} – error” (A.2.143) provides the error which represents the uncertainty of the measured signal “Position uv_origin {x, y, z}” (A.2.142). |      |  |
| Value type  | scalar/vector/matrix real value<br>(see B.4.1)   | Unit | See signal “Position uv_origin {x, y, z}” (A.2.142). |

### A.2.144 Position u\_end {x, y, z}

**Table A.251 — Signal: Position u\_end {x, y, z}**

| Name | Position u_end {x, y, z} |
|------|--------------------------|
|------|--------------------------|

|                    |   |             |           |
|--------------------|---|-------------|-----------|
| <b>Description</b> | <p>The signal "Position u_end {x, y, z}" (A.2.144) provides the position of the reference corner of a sign board which defines the one end of a single sign board affine object coordinate system in horizontal direction. The point is defined in the ego-vehicle coordinate system [see signal "Vehicle coordinate system type – header" (A.1.20)] (see Figure A.16).</p> <p>Additional information:</p> <p>Even though not always all corners are visible, the origin "Position uv_origin {x, y, z}" (A.2.142) shall be always the left lower corner. The direction of the affine coordinate system is defined by the end positions "Position u_end {x, y, z}" (A.2.144) in negative Y direction of the ego-vehicle coordinate system and "Position v_end {x, y, z}" (A.2.146) in the positive Z direction of the ego-vehicle coordinate system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p> |             |           |
| <b>Value type</b>  | 3D vector real value  | <b>Unit</b> | (m, m, m) |

#### A.2.145 Position u\_end {x, y, z} – error

**Table A.252 — Signal: Position u\_end {x, y, z} – error**

|                    |  |             |  |
|--------------------|--|-------------|--|
| <b>Name</b>        | Position u_end {x, y, z} – error   |             |  |
| <b>Description</b> | The signal "Position u_end {x, y, z} – error" (A.2.145) provides the error which represents the uncertainty of the measured signal "Position u_end {x, y, z}" (A.2.144). |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal "Position u_end {x, y, z}" (A.2.144). |

#### A.2.146 Position v\_end {x, y, z}

**Table A.253 — Signal: Position v\_end {x, y, z}**

|                    |   |             |           |
|--------------------|---|-------------|-----------|
| <b>Name</b>        | Position v_end {x, y, z}  |             |           |
| <b>Description</b> | <p>The signal "Position v_end {x, y, z}" (A.2.146) provides the position of the reference corner of a sign board which defines the one end of a single sign board affine object coordinate system in vertical direction. The point is defined in the ego-vehicle coordinate system [see signal "Vehicle coordinate system type – header" (A.1.20)] (see Figure A.16).</p> <p>Additional information:</p> <p>Even though not always all corners are visible, the origin "Position uv_origin {x, y, z}" (A.2.142) shall be always the left lower corner. The direction of the affine coordinate system is defined by the end positions "Position u_end {x, y, z}" (A.2.144) in negative Y direction of the ego-vehicle coordinate system and "Position v_end {x, y, z}" (A.2.146) in the positive Z direction of the ego-vehicle coordinate system [see signal "Vehicle coordinate system type – header" (A.1.20)].</p> |             |           |
| <b>Value type</b>  | 3D vector real value  | <b>Unit</b> | (m, m, m) |

#### A.2.147 Position v\_end {x, y, z} – error

**Table A.254 — Signal: Position v\_end {x, y, z} – error**

|                    |  |  |  |
|--------------------|--|--|--|
| <b>Name</b>        | Position v_end {x, y, z} – error   |  |  |
| <b>Description</b> | The signal "Position v_end {x, y, z} – error" (A.2.147) provides the error which represents the uncertainty of the measured signal "Position v_end {x, y, z}" (A.2.146). |  |  |

|                   |  |             |  |
|-------------------|--|-------------|--|
| <b>Value type</b> | scalar/vector/matrix real value<br>(see B.4.1) | <b>Unit</b> | See signal “Position v_end {x, y, z}” (A.2.146). |
|-------------------|--|-------------|--|

**A.2.148 Ratio U- to V-axis****Table A.255 — Signal: Ratio U- to V-axis**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Ratio U- to V-axis   |             |   |
| <b>Description</b> | <p>The signal “Ratio U- to V-axis” (A.2.148) provides the ratio of the scaling of the <i>U</i>-axis and the <i>V</i>-axis unit vectors length in the cartesian coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)].</p> <p>Additional information:</p> <p>The signal “Ratio U- to V-axis” (A.2.148) <math>\doteq</math> (length of the unit vector of <i>U</i>-axis in m)/(length of the unit vector of <i>V</i>-axis in m).</p> |             |   |
| <b>Value type</b>  | real value   | <b>Unit</b> | 1 |

**A.2.149 Ratio U- to V-axis – error****Table A.256 — Signal: Ratio U- to V-axis – error**

|                    |  |             |  |
|--------------------|--|-------------|--|
| <b>Name</b>        | Ratio U- to V-axis – error   |             |  |
| <b>Description</b> | The signal “Ratio U- to V-axis – error” (A.2.149) provides the error which represents the uncertainty of the measured signal “Ratio U- to V-axis” (A.2.148). |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Ratio U- to V-axis” (A.2.148). |

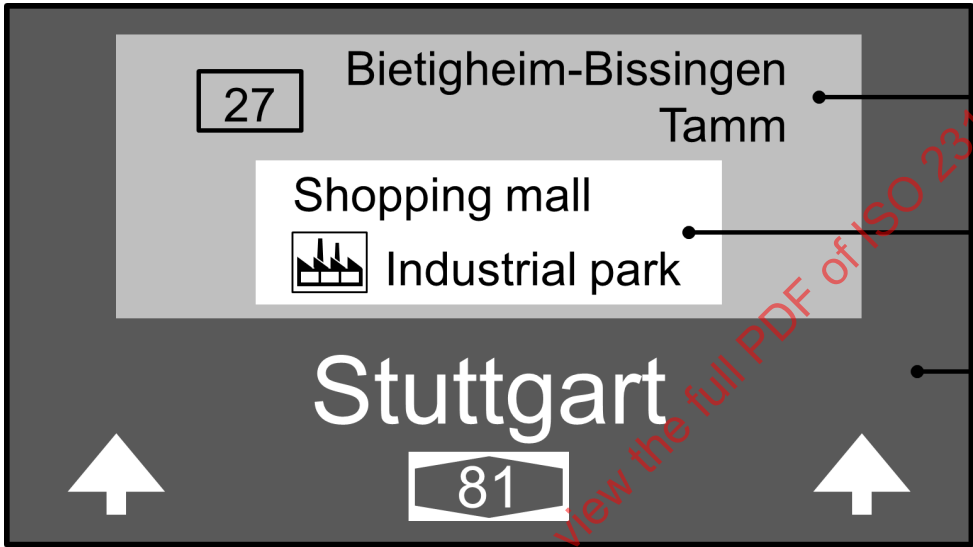
**A.2.150 Number of valid traffic sign board regions****Table A.257 — Signal: Number of valid traffic sign board regions**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid traffic sign board regions   |             |   |
| <b>Description</b> | <p>The signal “Number of valid traffic sign board regions” (A.2.150) provides the current number of valid traffic sign board regions.</p> <p>Additional information:</p> <p>The entries in the array may or may not be sorted.</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.2.151 Sign board entity ID****Table A.258 — Signal: Sign board entity ID**

|             |                      |
|-------------|----------------------|
| <b>Name</b> | Sign board entity ID |
|-------------|----------------------|

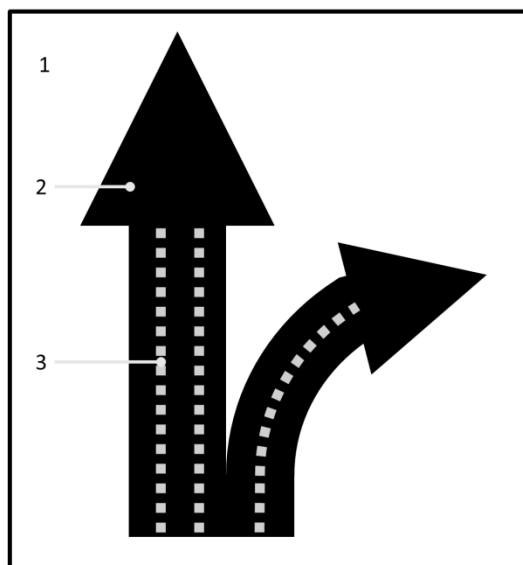
|             |  |      |   |
|-------------|--|------|---|
| Description | The signal “Sign board entity ID” (A.2.151) provides the unique ID of a sign board type entity within the sign board object (see Figure A.17 and Figure A.18). |      |   |
|             | Additional information:  |      |   |
|             | The signal’s “Sign board entity ID” (A.2.151) value is unique for the same entity in multiple cycles.  |      |   |
|             | The signal “Sign board entity ID” (A.2.151) := 0 refers to the traffic sign board entity.  |      |   |
| Value type  | integer value  | Unit | 1 |



Key

- 1 traffic sign board – motorway  
[Default: “Sign board entity ID” (A.2.151) := 0]
- 2 traffic sign board region – rural  
signal “Sign board entity ID” (A.2.151) := 1 and signal “Sign board entity ID reference – object level” (A.2.153) := {0}
- 3 traffic sign board region - city  
signal “Sign board entity ID” (A.2.151) := 2 and signal “Sign board entity ID reference – object level” (A.2.153) := {1}

Figure A.17 — Example for a logical hierarchy of traffic sign board region entities

**Key**

- 1 traffic sign board with an orientation arrow of two directions and five lanes  
[default: "Sign board entity ID" (A.2.151) := 0]
- 2 traffic sign board guiding graph – orientation arrow  
signal "Sign board entity ID" (A.2.151) := 1 and signal "Sign board entity ID reference – object level" (A.2.153) := {0}
- 3 traffic sign board guiding graph – lanes  
signal "Sign board entity ID" (A.2.151) := 2 and signal "Sign board entity ID reference – object level" (A.2.153) := {1}

**Figure A.18 — Example for a logical hierarchy of traffic sign board guiding graph entities****A.2.152 Number of valid entity ID references – object level****Table A.259 — Signal: Number of valid entity ID references – object level**

| Name        | Number of valid entity ID references – object level   |      |   |
|-------------|---|------|---|
| Description | The signal "Number of valid entity ID references – object level" (A.2.152) provides the current number of valid traffic sign board type entities (for example, region, referenced object, symbol or icon) on the traffic sign board. All referenced object entities are associated with the traffic sign board. |      |   |
|             | Additional information:<br>The entries in the array may or may not be sorted.   |      |   |
| Value type  | [0...] integer value  | Unit | 1 |

**A.2.153 Sign board entity ID reference – object level****Table A.260 — Signal: Sign board entity ID reference – object level**

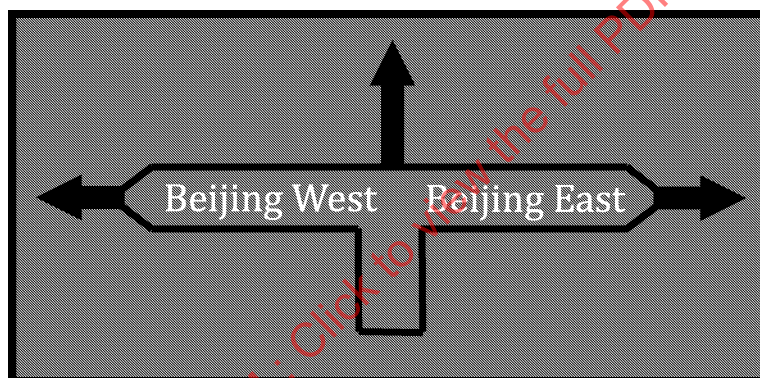
| Name        | Sign board entity ID reference – object level   |
|-------------|---|
| Description | The signal's "Sign board entity ID" (A.2.151) value defines a sign board type entity to which this entity is logical associated to (see Figure A.17 and Figure A.18). |
|             | Additional information:<br>The signal "Sign board entity ID" (A.2.151) := 0 refers to the traffic sign board entity itself.   |

|                   |                      |             |   |
|-------------------|----------------------|-------------|---|
| <b>Value type</b> | [0...] integer value | <b>Unit</b> | 1 |
|-------------------|----------------------|-------------|---|

#### A.2.154 Border width {u, v}

**Table A.261 — Signal: Border width {u, v}**

|                    |  |             |        |
|--------------------|--|-------------|--------|
| <b>Name</b>        | Border width {u, v}  |             |        |
| <b>Description</b> | <p>The signal “Border width {u, v}” (A.2.154) provides a borderline width of the edge scaled to the <i>U</i>- and <i>V</i>-axis (see Figure A.19).</p> <p>Additional information:</p> <p>The scaled width {u} is relative to the <i>U</i>-axis of the current measurement cycle and the scaled height {v} is relative to the <i>V</i>-axis of the current measurement cycle.</p> <p>As a horizontal line cannot have a width in <i>U</i>-direction and a vertical line cannot have a width in <i>V</i>-direction, the border width {u} shall be used for the vertical parts of the border and the border width {v} shall be used for the horizontal parts of the border.</p> |             |        |
| <b>Value type</b>  | 2D vector [0...1] real value   | <b>Unit</b> | (1, 1) |



**Figure A.19 — Example of a traffic sign board with text on guiding graph which has a borderline**

#### A.2.155 Border width {u, v} - error

**Table A.262 — Signal: Border width {u, v} - error**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Border width {u, v} - error  |             |   |
| <b>Description</b> | The signal “Border width {u, v} - error” (A.2.155) provides the error which represents the uncertainty of the measured signal “Border width {u, v}” (A.2.154). |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value (see B.4.1)  | <b>Unit</b> | See signal “Border width {u, v}” (A.2.154). |

#### A.2.156 Traffic sign board region - confidence

**Table A.263 — Signal: Traffic sign board region - confidence**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Traffic sign board region - confidence  |             |   |
| <b>Description</b> | The signal “Traffic sign board region - confidence” (A.2.156) provides the confidence for the region. |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

**A.2.157 Number of valid colour tones****Table A.264 — Signal: Number of valid colour tones**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid colour tones   |             |   |
| <b>Description</b> | <p>The signal “Number of valid colour tones” (A.2.157) provides the current number of valid colour tones for one colour tone type.</p> <p>Additional information:<br/>The entries in the array may or may not be sorted.</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.2.158 Colour tone type****Table A.265 — Signal: Colour tone type**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Colour tone type  |             |   |
| <b>Description</b> | The signal “Colour tone type” (A.2.158) provides the type of the colour tone. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.266 — Enumeration: Colour tone type**

| <b>Name</b> | <b>Description</b>                                    | <b>RL enumerator</b> |
|-------------|---|----------------------|
| CTT_Shape   | The colour tone defines the colour of the shape.      | Mandatory            |
| CTT_Border  | The colour tone defines the colour of the borderline. | Optional             |

**A.2.159 Number of valid vertex points****Table A.267 — Signal: Number of valid vertex points**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid vertex points  |             |   |
| <b>Description</b> | <p>The signal “Number of valid vertex points” (A.2.159) provides the current number of valid vertex points of the traffic sign board entity.</p> <p>Additional information:<br/>The entries in the array may or may not be sorted.</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.2.160 Local position {u, v}****Table A.268 — Signal: Local position {u, v}**

|                    |  |  |  |
|--------------------|--|--|--|
| <b>Name</b>        | Local position {u, v}  |  |  |
| <b>Description</b> | <p>The signal “Local position {u, v}” (A.2.160) provides a point on the traffic sign board in the traffic sign board affine coordinate system.</p> <p>Additional information:<br/>The origin point of the traffic sign board affine coordinate system is in the lower left corner of the traffic sign board (from point of view of the ego-vehicle coordinate system). The <i>U</i>-axis is horizontal, and the <i>V</i>-axis is vertical.</p> |  |  |

|                   |                              |             |        |
|-------------------|------------------------------|-------------|--------|
| <b>Value type</b> | 2D vector [0...1] real value | <b>Unit</b> | (1, 1) |
|-------------------|------------------------------|-------------|--------|

### A.2.161 Local position $\{u, v\}$ - error

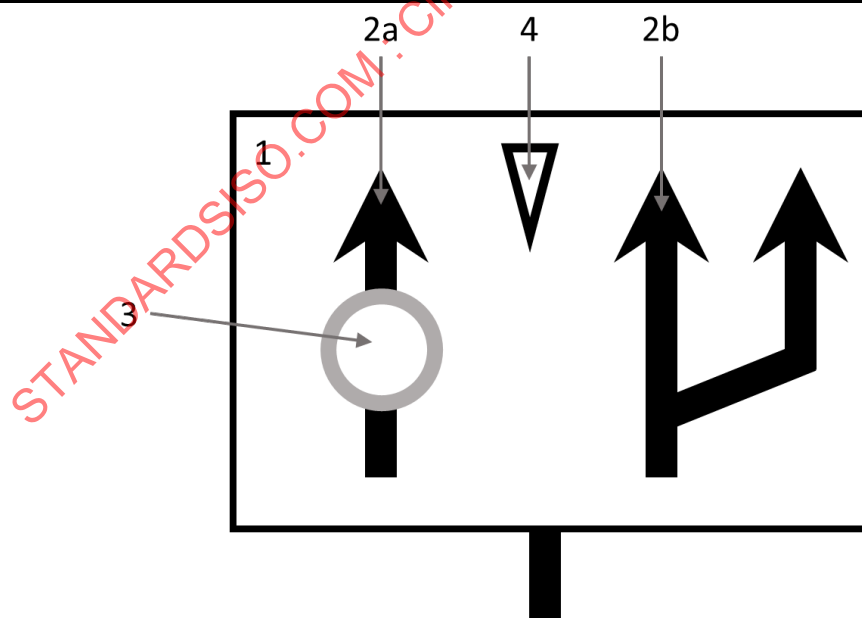
**Table A.269 — Signal: Local position {u, v} – error**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Local position {u, v} – error  |             |   |
| <b>Description</b> | The signal “Local position {u, v} – error” (A.2.161) provides the error which represents the uncertainty of the measured signal “Local position {u, v}” (A.2.160). |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Local position {u, v}” (A.2.160). |

#### A.2.162 Number of valid traffic sign board guiding graphs

**Table A.270 — Signal: Number of valid traffic sign board guiding graphs**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid traffic sign board guiding graphs  |             |   |
| <b>Description</b> | <p>The signal “Number of valid traffic sign board guiding graphs” (A.2.162) provides the current number of valid guiding graphs (for example, orientation arrow, marking of a shoulder) on the traffic sign board (see Figure A.20).</p> <p>Additional information:</p> <p>The entries in the array may or may not be sorted.</p> <p>A guiding graph may describe separated direction arrows in one entity.</p> <p>A sensor cluster may describe all guiding graphs in one or multiple entities of the traffic sign board.</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |



### Key

- 1 traffic sign board  
2a,b orientation arrow symbol – guiding graph(s)  
(the orientation arrows may be described by one entity or separated in different entities)  
3 object reference – traffic sign as traffic main sign



(signal “Object ID reference – object level” (A.2.179) references to a traffic sign as traffic main sign which describes the sign)

4 icons and symbols – a road separation symbol

(the signal “Icon and symbol type” (A.2.176) has an enumerator “IAST\_RoadSeparation”)

**Figure A.20 — Example for a traffic sign board with guiding graph(s)**

#### A.2.163 Traffic sign board guiding graph – confidence

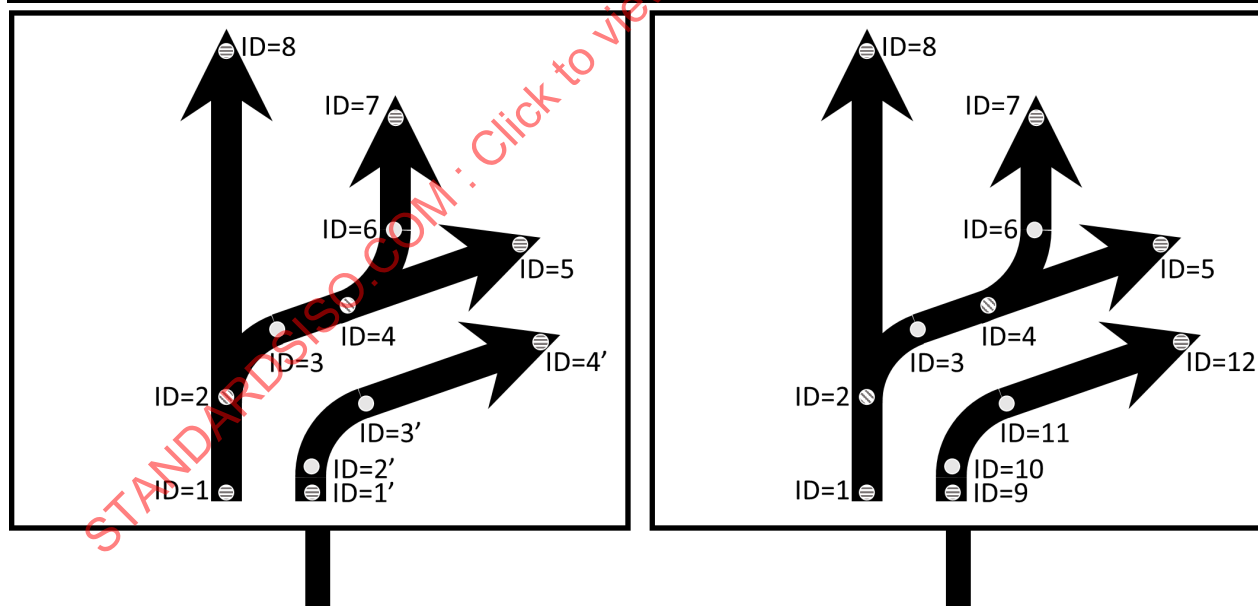
**Table A.271 — Signal: Traffic sign board guiding graph – confidence**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Traffic sign board guiding graph – confidence   |             |   |
| <b>Description</b> | The signal “Traffic sign board guiding graph – confidence” (A.2.163) provides the confidence for the guiding graph. |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

#### A.2.164 Local position ID

**Table A.272 — Signal: Local position ID**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Local position ID  |             |   |
| <b>Description</b> | The signal “Local position ID” (A.2.164) provides a unique ID for a signal “Local position {u, v}” (A.2.160) within the list of local positions (see Figure A.21). |             |   |
| <b>Value type</b>  | integer value  | <b>Unit</b> | 1 |



#### Key

- begin or end vertex point
- split or merge vertex point
- intermediate vertex point

ID= ID of the vertex point – signal “Local position ID” (A.2.164)

**Figure A.21 — Two examples for traffic sign board guiding graphs and their vertex point IDs “Local position ID” (A.2.164)**

A.2.165 Number of valid edges

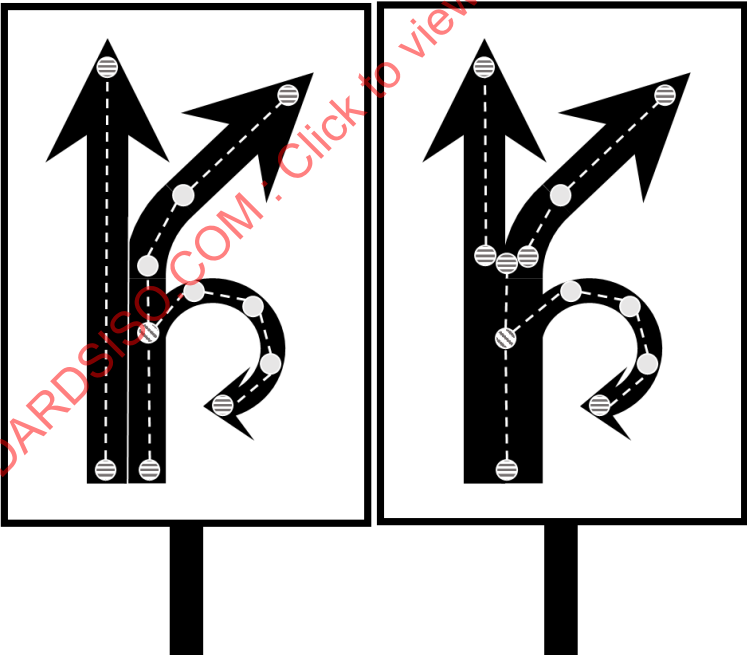
Table A.273 — Signal: Number of valid edges

|             |   |      |   |
|-------------|---|------|---|
| Name        | Number of valid edges   |      |   |
| Description | The signal “Number of valid edges” (A.2.165) provides the current number of valid edges.<br><br>Additional information:<br>The entries in the array may or may not be sorted. |      |   |
| Value type  | [0...] integer value  | Unit | 1 |

A.2.166 Local position ID reference {begin, end}

Table A.274 — Signal: Local position ID reference {begin, end}

|             |  |      |        |
|-------------|--|------|--------|
| Name        | Local position ID reference {begin, end}   |      |        |
| Description | The signal “Local position ID reference {begin, end}” (A.2.166) provides an edge by connecting a {begin} vertex and an {end} vertex. Each vertex points can be every kind of type (begin-, end-, split-, merge vertex) (see Figure A.22).<br><br>Additional information:<br>The order of {begin} vertex and {end} vertex represents a direction. |      |        |
| Value type  | 2D vector integer value  | Unit | (1, 1) |



- Key
- begin or end vertex point
  - split or merge vertex point
  - intermediate vertex point
  - .... edge of a guiding graph

Figure A.22 — Two examples for a traffic sign board guiding graphs with their vertex points and edges

## A.2.167 Edge type

Table A.275 — Signal: Edge type

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Edge type   |             |   |
| <b>Description</b> | The signal “Edge type” (A.2.167) indicates the type of the guiding graph’s edge type. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

Table A.276 — Enumeration: Edge type – Example enumerators

| <b>Name</b>   | <b>Description</b>                                  | <b>RL enumerator</b> |
|---------------|---|----------------------|
| ET_Solid      | The edge of the guiding graph is a solid line.      | “exemplary”          |
| ET_Dashed     | The edge of the guiding graph is a dashed line.     | “exemplary”          |
| ET_Triangular | The edge of the guiding graph is a triangular line. | “exemplary”          |
| ET_Dotted     | The edge of the guiding graph is a dotted line.     | “exemplary”          |

## A.2.168 Edge width {u, v}

Table A.277 — Signal: Edge width {u, v}

|                    |   |             |        |
|--------------------|---|-------------|--------|
| <b>Name</b>        | Edge width {u, v}   |             |        |
| <b>Description</b> | <p>The signal “Edge width {u, v}” (A.2.168) provides a line width of the guiding graph’s edge scaled to the <i>U</i>- and <i>V</i>-axis.</p> <p>Additional information:</p> <p>The scaled width {u} is relative to the unit vector of the <i>U</i>-axis and the scaled height {v} is relative to the unit vector of the <i>V</i>-axis.</p> <p>To enable a correct representation of tapering lines, the border width shall be represented per edge.</p> <p>As a horizontal line cannot have a width in <i>U</i>-direction, border width {u} is per definition equal to 0. Same applies for width {v} in vertical <i>V</i>-direction. Only lines not parallel to both axis <i>U</i> and <i>V</i> have values unequal to zero for both dimensions {u, v}.</p> <p>If not specified differently, the width shall be measured at the {begin} vertex [see signal “Local position ID reference {begin, end}” (A.2.166)].</p> |             |        |
| <b>Value type</b>  | 2D vector [0...1] real value  | <b>Unit</b> | (1, 1) |

## A.2.169 Edge width {u, v} – error

Table A.278 — Signal: Edge width {u, v} – error

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Edge width {u, v} – error  |             |   |
| <b>Description</b> | The signal “Edge width {u, v} – error” (A.2.169) provides the error which represents the uncertainty of the measured signal “Edge width {u, v}” (A.2.168). |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Edge width {u, v}” (A.2.168). |

## A.2.170 Edge curve type

Table A.279 — Signal: Edge curve type

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Edge curve type   |             |   |
| <b>Description</b> | <p>The signal “Edge curve type” (A.2.170) describes the shape of the edge between two vertices. The direction of the edge is implicit defined by the begin and the end vertex [see signal “Local position ID reference {begin, end}” (A.2.166)] and defines, for example, left and right.</p> <p>Additional information:</p> <p>It is up to the sensors capability to use as many vertices as needed to describe a guiding graph on a traffic sign board. To ease the sampling of vertices, this signal offers a simplification by defining a set of standard paths (for example, S-curve) between to vertices.</p> |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.280 — Enumeration: Edge curve type – Example enumerators**

| Name           | Description                       | RL enumerator |
|----------------|-----------------------------------|---------------|
| ECT_Straight   | The edge is straight.             | “exemplary”   |
| ECT_SCurve     | The edge is an S-curve.           | “exemplary”   |
| ECT_ZCurve     | The edge is a Z-curve.            | “exemplary”   |
| ECT_RightCurve | The edge is a curve to the right. | “exemplary”   |
| ECT_LeftCurve  | The edge is a curve to the left.  | “exemplary”   |
| ECT_RightBend  | The edge is a bend to the right.  | “exemplary”   |
| ECT_LeftBend   | The edge is a bend to the left.   | “exemplary”   |

**A.2.171 Arrow type****Table A.281 — Signal: Arrow type**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Arrow type   |             |   |
| <b>Description</b> | The signal “Arrow type” (A.2.171) indicates an arrow at {begin} or {end} vertex point of the edge [see signal “Local position ID reference {begin, end}” (A.2.166)]. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.282 — Enumeration: Arrow type – Example enumerators**

| Name                        | Description  | RL enumerator |
|-----------------------------|--|---------------|
| AT_NoArrow                  | The symbol has no arrow or termination at {begin} or {end} vertex point of the edge. | “exemplary”   |
| AT_ArrowAtBeginPoint        | The symbol has an arrow at {begin} vertex point of the edge.                         | “exemplary”   |
| AT_ArrowAtEndPoint          | The symbol has an arrow at {end} vertex point of the edge.                           | “exemplary”   |
| AT_ArrowsAtBothPoints       | The symbol has an arrow at {begin} and {end} vertex point of the edge.               | “exemplary”   |
| AT_TerminationAtBeginPoint  | The symbol has a termination at {begin} vertex point of the edge.                    | “exemplary”   |
| AT_TerminationAtEndPoint    | The symbol has a termination at {end} vertex point of the edge.                      | “exemplary”   |
| AT_TerminationsAtBothPoints | The symbol has a termination at {begin} and {end} vertex point of the edge.          | “exemplary”   |

**A.2.172 Number of valid traffic sign board texts****Table A.283 — Signal: Number of valid traffic sign board texts**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid traffic sign board texts  |             |   |
| <b>Description</b> | <p>The signal “Number of valid traffic sign board texts” (A.2.172) provides the current number of valid text fragments on the traffic sign board.</p> <p>Additional information:<br/>The entries in the array may or may not be sorted.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.2.173 Bounding box {u<sub>Begin</sub>, u<sub>End</sub>, v<sub>Begin</sub>, v<sub>End</sub>}****Table A.284 — Signal: Bounding box {u<sub>Begin</sub>, u<sub>End</sub>, v<sub>Begin</sub>, v<sub>End</sub>}**

|                    |  |             |              |
|--------------------|--|-------------|--------------|
| <b>Name</b>        | Bounding box {u <sub>Begin</sub> , u <sub>End</sub> , v <sub>Begin</sub> , v <sub>End</sub> }  |             |              |
| <b>Description</b> | <p>The signal “Bounding box {u<sub>Begin</sub>, u<sub>End</sub>, v<sub>Begin</sub>, v<sub>End</sub>}” (A.2.173) provides a bounding box on the traffic sign board (and defines the affine traffic sign board coordinate system) for a traffic sign board entity (for example, a referenced traffic sign or traffic light object entity) (see Figure A.16).</p> |             |              |
| <b>Value type</b>  | 4D vector [0...1] real value   | <b>Unit</b> | (1, 1, 1, 1) |

**A.2.174 Bounding box {u<sub>Begin</sub>, u<sub>End</sub>, v<sub>Begin</sub>, v<sub>End</sub>} – error****Table A.285 — Signal: Bounding box {u<sub>Begin</sub>, u<sub>End</sub>, v<sub>Begin</sub>, v<sub>End</sub>} – error**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Bounding box {u <sub>Begin</sub> , u <sub>End</sub> , v <sub>Begin</sub> , v <sub>End</sub> } – error   |             |   |
| <b>Description</b> | <p>The signal “Bounding box {u<sub>Begin</sub>, u<sub>End</sub>, v<sub>Begin</sub>, v<sub>End</sub>} – error” (A.2.174) provides the error which represents the uncertainty of the measured signal “Bounding box {u<sub>Begin</sub>, u<sub>End</sub>, v<sub>Begin</sub>, v<sub>End</sub>}” (A.2.173).</p> |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal “Bounding box {u <sub>Begin</sub> , u <sub>End</sub> , v <sub>Begin</sub> , v <sub>End</sub> }” (A.2.173). |

**A.2.175 Number of valid traffic sign board icons and symbols****Table A.286 — Signal: Number of valid traffic sign board icons and symbols**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid traffic sign board icons and symbols   |             |   |
| <b>Description</b> | <p>The signal “Number of valid traffic sign board icons and symbols” (A.2.175) provides the current number of valid icons and symbols on the traffic sign board.</p> <p>Additional information:<br/>The entries in the array may or may not be sorted.</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.2.176 Icon and symbol type****Table A.287 — Signal: Icon and symbol type**

|             |                      |
|-------------|----------------------|
| <b>Name</b> | Icon and symbol type |
|-------------|----------------------|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | The signal “Icon and symbol type” (A.2.176) indicates the type of the icon or the symbol of a commonly used symbol for characteristic road elements (for example, road separation) or destinations (for example, city centre, central station) but no traffic sign main or supplementary signs. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.288 — Enumeration: Icon and symbol type – Example enumerators**

| <b>Name</b>                | <b>Description</b>   | <b>RL enumerator</b> |
|----------------------------|--|----------------------|
| IAST_CentralStation        | The icon symbolises the central station.   | “exemplary”          |
| IAST_Airport               | The icon symbolises the airport.   | “exemplary”          |
| IAST_AirportDeparture      | The icon symbolises the departure at the airport.                                  | “exemplary”          |
| IAST_AirportArrival        | The icon symbolises the arrival at the airport.                                    | “exemplary”          |
| IAST_Church                | The icon symbolises a church.  | “exemplary”          |
| IAST_CityCentre            | The icon symbolises the city centre.   | “exemplary”          |
| IAST_IndustrialPark        | The icon symbolises an industrial park.  | “exemplary”          |
| IAST_Plant                 | The icon symbolises a plant.   | “exemplary”          |
| IAST_Sightseeing           | The icon symbolises sight-seeing.  | “exemplary”          |
| IAST_RoadSeparation        | The symbol symbolises a separation of the road to guide single or multiple lanes.  | “exemplary”          |
| IAST_Harbour               | The icon symbolises a harbour.   | “exemplary”          |
| IAST_Hospital              | The icon symbolises a hospital or a medical service.                               | “exemplary”          |
| IAST_Stadium               | The icon symbolises a stadium.   | “exemplary”          |
| IAST_Police                | The icon symbolises a police station.  | “exemplary”          |
| IAST_Cemetery              | The icon symbolises a cemetery.  | “exemplary”          |
| IAST_Mining                | The icon symbolises a mining.  | “exemplary”          |
| IAST_InformationPoint      | The icon symbolises an information point.  | “exemplary”          |
| IAST_Toll                  | The icon symbolises a toll station.  | “exemplary”          |
| IAST_GasStation            | The icon symbolises a gas station.   | “exemplary”          |
| IAST_ElectricChargingPoint | The icon symbolises an electric charging point.                                    | “exemplary”          |
| IAST_Tunnel                | The icon symbolises a tunnel portal.   | “exemplary”          |
| IAST_Bridge                | The icon symbolises a bridge.  | “exemplary”          |
| IAST_UnrecognisedSymbol    | The symbol is not a recognised symbol or icon, for example, a locally used symbol. | “exemplary”          |

**A.2.177 Icon and symbol type – confidence****Table A.289 — Signal: Icon and symbol type – confidence**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Icon and symbol type – confidence  |             |   |
| <b>Description</b> | The signal “Icon and symbol type – confidence” (A.2.177) provides the confidence for the icon and symbol type signal “Icon and symbol type” (A.2.176). |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

**A.2.178 Number of valid traffic sign board object references****Table A.290 — Signal: Number of valid traffic sign board object references**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid traffic sign board object references   |             |   |
| <b>Description</b> | <p>The signal “Number of valid traffic sign board object references” (A.2.178) provides the current number of valid references to object entities on the traffic sign board which could be described in a separate object entity (for example, traffic sign, traffic light).</p> <p>Additional information:<br/>The entries in the array may or may not be sorted.</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.2.179 Object ID reference – object level****Table A.291 — Signal: Object ID reference – object level**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Object ID reference – object level  |             |   |
| <b>Description</b> | <p>The signal’s “Object ID” (A.2.2) value defines the object to which this object is associated to; invalid = “no associated object”.</p> <p>Additional information:<br/>Refers to an object entity with the highest probability for the association, for example, a free-space area references a static-object entity which limits the area, a traffic sign board references a traffic sign which is associated to the traffic sign board.<br/>The signal’s “Object ID reference – object level” (A.2.179) value could reference entities of the sensor cluster’s OLI or entities which are used internally by the sensor cluster.<br/>Alternative A2I (B.3.2): the mapping of objects signal “Object ID” (A.2.2) and other object level interfaces (for example, free space area object and the object’s limitations) signal “Object ID reference – feature level” (A.3.3) could be provided in a separate interface.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.2.180 Object ID reference – confidence****Table A.292 — Signal: Object ID reference – confidence**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Object ID reference – confidence   |             |   |
| <b>Description</b> | <p>The signal “Object ID reference – confidence” (A.2.180) provides the confidence for the object reference signal “Object ID reference – object level” (A.2.179).</p> |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

**A.2.181 Number of valid traffic sign board occlusions****Table A.293 — Signal: Number of valid traffic sign board occlusions**

|                    |  |  |  |
|--------------------|--|--|--|
| <b>Name</b>        | Number of valid traffic sign board occlusions  |  |  |
| <b>Description</b> | <p>The signal “Number of valid traffic sign board occlusions” (A.2.181) provides the current number of valid occlusion parts on or in front of the traffic sign board.</p> <p>Additional information:<br/>The entries in the array may or may not be sorted.</p> |  |  |

|                   |                      |             |   |
|-------------------|----------------------|-------------|---|
| <b>Value type</b> | [0...] integer value | <b>Unit</b> | 1 |
|-------------------|----------------------|-------------|---|

### A.2.182 Occlusion type

**Table A.294 — Signal: Occlusion type**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Occlusion type   |             |   |
| <b>Description</b> | The signal “Occlusion type” (A.2.182) indicates the type of occlusion of traffic sign board parts. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.295 — Enumeration: Occlusion type - Example enumerators**

| <b>Name</b>      | <b>Description</b>   | <b>RL enumerator</b> |
|------------------|--|----------------------|
| OT_Graffiti      | Parts of the traffic sign boards are occluded by graffiti or stickers.                                       | “exemplary”          |
| OT_Vegetation    | Parts of the traffic sign boards are occluded by vegetation, for example tree branches, leaves or the trunk. | “exemplary”          |
| OT_MovingObjects | Parts of the traffic sign boards are occluded by moving objects, for example, a heavy truck.                 | “exemplary”          |
| OT_Snow          | Parts of the traffic sign boards are occluded by snow.   | “exemplary”          |
| OT_Dirt          | Parts of the traffic sign boards are occluded by dirt.   | “exemplary”          |

### A.2.183 Occlusion type - confidence

**Table A.296 — Signal: Occlusion type - confidence**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Occlusion type - confidence  |             |   |
| <b>Description</b> | The signal “Occlusion type - confidence” (A.2.183) provides the confidence for the occlusion type signal “Occlusion type” (A.2.182). |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

### A.2.184 Number of valid structure light classifications

**Table A.297 — Signal: Number of valid structure light classifications**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid structure light classifications  |             |   |
| <b>Description</b> | The signal “Number of valid structure light classifications” (A.2.184) provides the current number of valid tuples for structure classifications of the traffic light. The tuples are defined by the signals “Structure light classification type” (A.2.185) and “Structure light classification type - confidence” (A.2.186). |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

### A.2.185 Structure light classification type

**Table A.298 — Signal: Structure light classification type**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Structure light classification type  |             |   |
| <b>Description</b> | The signal “Structure light classification type” (A.2.185) provides the classification of the traffic light’s shape. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |



**Table A.299 — Enumeration: Structure light classification type – Example enumerators**

| Name             | Description  | RL enumerator |
|------------------|--|---------------|
| SLCT_Vertical3   | The traffic light is composed of three vertical light spots.   | “exemplary”   |
| SLCT_Horizontal3 | The traffic light is composed of three horizontal light spots. | “exemplary”   |
| SLCT_DogHouse    | The traffic light is composed of multiple light spots.         | “exemplary”   |

**A.2.186 Structure light classification type – confidence****Table A.300 — Signal: Structure light classification type – confidence**

| Name        | Structure light classification type – confidence   |      |   |
|-------------|--|------|---|
| Description | The signal “Structure light classification type – confidence” (A.2.186) provides the confidence for the signal’s “Structure light classification type” (A.2.185) classification information, for example, the degree of certainty that the structure light type is of the reported type, for example, a traffic light with 3 vertical stacked traffic light spots. |      |   |
| Value type  | [0...100] real value   | Unit | % |

**A.2.187 Minimum visibility distance****Table A.301 — Signal: Minimum visibility distance**

| Name        | Minimum visibility distance  |      |   |
|-------------|--|------|---|
| Description | The signal “Minimum visibility distance” (A.2.187) provides the minimum distance to the traffic light until the traffic light is still in the FOV of the sensor cluster, for example, in close distance. |      |   |
| Value type  | [0...] real value  | Unit | m |

**A.2.188 Total number of traffic light spots****Table A.302 — Signal: Total number of traffic light spots**

| Name        | Total number of traffic light spots  |      |   |
|-------------|--|------|---|
| Description | <p>The signal “Total number of traffic light spots” (A.2.188) provides the estimation of the number of traffic light spots which are part of the traffic light.</p> <p>Additional information:</p> <p>This signal is considered to be an estimation since turned-off spots are barely visible (especially at night) and therefore, hard to detect.</p> <p>Signal “Total number of traffic light spots” (A.2.188) <math>\geq</math> signal “Number of valid traffic light spots” (A.2.190).</p> |      |   |
| Value type  | [0...] integer value   | Unit | 1 |

**A.2.189 Total number of traffic light spots – confidence****Table A.303 — Signal: Total number of traffic light spots – confidence**

| Name | Total number of traffic light spots – confidence |  |  |
|------|--|--|--|
|------|--|--|--|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | The signal “Total number of traffic light spots – confidence” (A.2.189) provides the confidence for the signal’s “Total number of traffic light spots” (A.2.188) classification information, for example, the degree of certainty that the number of traffic light spots is of the reported type, for example, 3. |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

### A.2.190 Number of valid traffic light spots

**Table A.304 — Signal: Number of valid traffic light spots**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid traffic light spots  |             |   |
| <b>Description</b> | The signal “Number of valid traffic light spots” (A.2.190) provides the current number of valid traffic light entities.<br><br>Additional information:<br>The entries in the array may or may not be sorted. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

### A.2.191 Number of valid light shape classifications

**Table A.305 — Signal: Number of valid light shape classifications**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid light shape classifications   |             |   |
| <b>Description</b> | The signal “Number of valid light shape classifications” (A.2.191) provides the current number of valid tuples for light shape classifications. The tuples are defined by the signals “Light shape classification type” (A.2.192) and “Light shape classification type – confidence” (A.2.193). |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

### A.2.192 Light shape classification type

**Table A.306 — Signal: Light shape classification type**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Light shape classification type   |             |   |
| <b>Description</b> | The signal “Light shape classification type” (A.2.192) provides the light’s shape which is displayed by the light spot. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.307 — Enumeration: Light shape classification type – Example enumerators**

| <b>Name</b>                 | <b>Description</b>  | <b>RL enumerator</b> |
|-----------------------------|---|----------------------|
| LSCT_NoShape                | The traffic light spot displays no additional shape.                          | “exemplary”          |
| LSCT_ArrowStraightAhead     | The traffic light spot displays an arrow straight ahead shape.                | “exemplary”          |
| LSCT_ArrowLeft              | The traffic light spot displays an arrow left shape.                          | “exemplary”          |
| LSCT_ArrowDiagonalLeft      | The traffic light spot displays an arrow diagonal left shape.                 | “exemplary”          |
| LSCT_ArrowStraightAheadLeft | The traffic light spot displays an arrow straight ahead and arrow left shape. | “exemplary”          |
| LSCT_ArrowRight             | The traffic light spot displays an arrow right shape.                         | “exemplary”          |

| Name                         | Description   | RL enumerator |
|------------------------------|---|---------------|
| LSCT_ArrowDiagonalRight      | The traffic light spot displays an arrow diagonal right shape.  | "exemplary"   |
| LSCT_ArrowStraightAheadRight | The traffic light spot displays an arrow straight ahead and arrow right shape.  | "exemplary"   |
| LSCT_ArrowLeftRight          | The traffic light spot displays an arrow left and arrow right shape.  | "exemplary"   |
| LSCT_ArrowDown               | The traffic light spot displays an arrow down shape.  | "exemplary"   |
| LSCT_ArrowDownLeft           | The traffic light spot displays an arrow U-turn left shape.   | "exemplary"   |
| LSCT_ArrowDownRight          | The traffic light spot displays an arrow U-turn right shape.  | "exemplary"   |
| LSCT_Cross                   | The traffic light spot displays a cross figure.   | "exemplary"   |
| LSCT_Pedestrian              | The traffic light spot displays a pedestrian figure.  | "exemplary"   |
| LSCT_Walk                    | The traffic light spot displays a text "walk" figure.   | "exemplary"   |
| LSCT_DontWalk                | The traffic light spot displays a text "don't walk" figure.   | "exemplary"   |
| LSCT_Bicycle                 | The traffic light spot displays a bicycle figure.   | "exemplary"   |
| LSCT_PedestrianAndBicycle    | The traffic light spot displays a pedestrian and bicycle figure.  | "exemplary"   |
| LSCT_CountdownSecond         | The traffic light spot displays a countdown in seconds figure, the signal "Light shape value" (A.2.194) contains the value in seconds.    | "exemplary"   |
| LSCT_CountdownPercent        | The traffic light spot displays a countdown in percent figure, the signal "Light shape value" (A.2.194) contains the value in percentage. | "exemplary"   |
| LSCT_Train                   | The traffic light spot displays a train or tram figure.   | "exemplary"   |
| LSCT_Bus                     | The traffic light spot displays a bus figure.   | "exemplary"   |
| LSCT_BusAndTrain             | The traffic light spot displays a bus and train or tram figure.   | "exemplary"   |

### A.2.193 Light shape classification type - confidence

**Table A.308 — Signal: Light shape classification type - confidence**

| Name        | Light shape classification type - confidence   |      |   |
|-------------|--|------|---|
| Description | The signal "Light shape classification type - confidence" (A.2.193) provides the confidence for the signal's "Light shape classification type" (A.2.192) classification information, for example, the degree of certainty that the light shape is of the reported type, for example, the shape of a bus. |      |   |
| Value type  | [0...100] real value   | Unit | % |

### A.2.194 Light shape value

**Table A.309 — Signal: Light shape value**

| Name | Light shape value |
|------|-------------------|
|------|-------------------|

|                    |   |             |  |
|--------------------|---|-------------|--|
| <b>Description</b> | <p>The signal “Light shape value” (A.2.194) provides an additional countdown value of the light shape which is defined by the signal “Light shape classification type” (A.2.192).</p> <p>Additional information:</p> <p>The value meaning and the value unit depends on the enumerator of the signal “Light shape classification type” (A.2.192).</p> |             |  |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | <p>The unit is defined by the enumerator of the signal “Light shape classification type” (A.2.192).</p> <p>NOTE The signal “Light shape classification type” (A.2.192) has an enumerator “LSCT_CountdownSecond”, an enumerator “LSCT_CountdownPercent” or a similar enumerator defined during the system design phase.</p> |

#### A.2.195 Number of valid colour classifications

**Table A.310 — Signal: Number of valid colour classifications**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid colour classifications   |             |   |
| <b>Description</b> | <p>The signal “Number of valid colour classifications” (A.2.195) provides the current number of valid tuples for colour classifications. The tuples are defined by the signals “Colour classification type” (A.2.196) and “Colour classification type – confidence” (A.2.197).</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

#### A.2.196 Colour classification type

**Table A.311 — Signal: Colour classification type**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Colour classification type  |             |   |
| <b>Description</b> | <p>The signal “Colour classification type” (A.2.196) provides the colour of the light spot.</p> |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.312 — Enumeration: Colour classification type – Example enumerators**

| <b>Name</b> | <b>Description</b>               | <b>RL enumerator</b> |
|-------------|----------------------------------|----------------------|
| CCT_Red     | The light spot colour is red.    | “exemplary”          |
| CCT_Yellow  | The light spot colour is yellow. | “exemplary”          |
| CCT_Green   | The light spot colour is green.  | “exemplary”          |
| CCT_White   | The light spot colour is white.  | “exemplary”          |

#### A.2.197 Colour classification type – confidence

**Table A.313 — Signal: Colour classification type – confidence**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Colour classification type – confidence  |             |   |
| <b>Description</b> | <p>The signal “Colour classification type – confidence” (A.2.197) provides a confidence for the corresponding signal “Colour classification type” (A.2.196).</p> |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

#### A.2.198 Number of valid light mode classifications

**Table A.314 — Signal: Number of valid light mode classifications**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid light mode classifications  |             |   |
| <b>Description</b> | The signal "Number of valid light mode classifications" (A.2.198) provides the current number of valid tuples for light mode classifications. The tuples are defined by the signals "Light mode classification type" (A.2.199) and "Light mode classification type – confidence" (A.2.200). |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

### A.2.199 Light mode classification type

**Table A.315 — Signal: Light mode classification type**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Light mode classification type   |             |   |
| <b>Description</b> | The signal "Light mode classification type" (A.2.199) provides the light's mode. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.316 — Enumeration: Light mode classification type – Example enumerators**

| <b>Name</b>     | <b>Description</b>  | <b>RL enumerator</b> |
|-----------------|---|----------------------|
| LMCT_Continuous | The light source is currently on, and it is not blinking. | "exemplary"          |
| LMCT_Blinking   | One light source is visibly blinking.                     | "exemplary"          |
| LMCT_TurnedOff  | The light source is turned off.                           | "exemplary"          |
| LMCT_Counting   | It is a light source with counting.                       | "exemplary"          |

### A.2.200 Light mode classification type – confidence

**Table A.317 — Signal: Light mode classification type – confidence**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Light mode classification type – confidence   |             |   |
| <b>Description</b> | The signal "Light mode classification type – confidence" (A.2.200) provides a confidence for the corresponding signal "Light mode classification type" (A.2.199). |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

### A.2.201 Free space type

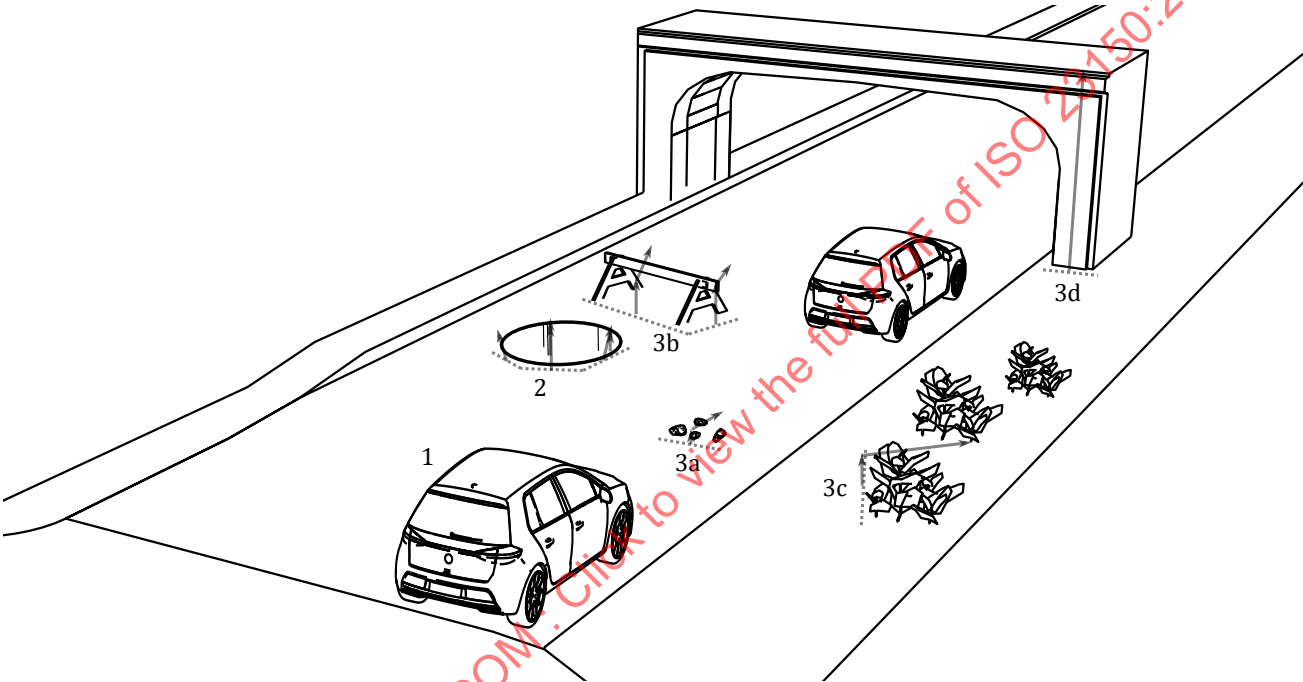
**Table A.318 — Signal: Free space type**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Free space type  |             |   |
| <b>Description</b> | The signal "Free space type" (A.2.201) describes the type of information that the free space description contains. This type of information depends on the recognition capabilities of the sensor (measurable states) and the combination of object properties that are required for object level fusion. The fusion might request for a free space description with or without static or dynamic objects or even holes ("negative limiting objects"). |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.319 — Enumeration: Free space type**

| <b>Name</b>         | <b>Description</b>  | <b>RL enumerator</b> |
|---------------------|---|----------------------|
| FST_LimitedByStatic | The space is limited by static objects, no confirmed free space area by measurement. Dynamic objects may not be recognized (see Figure A.23). | Mandatory            |

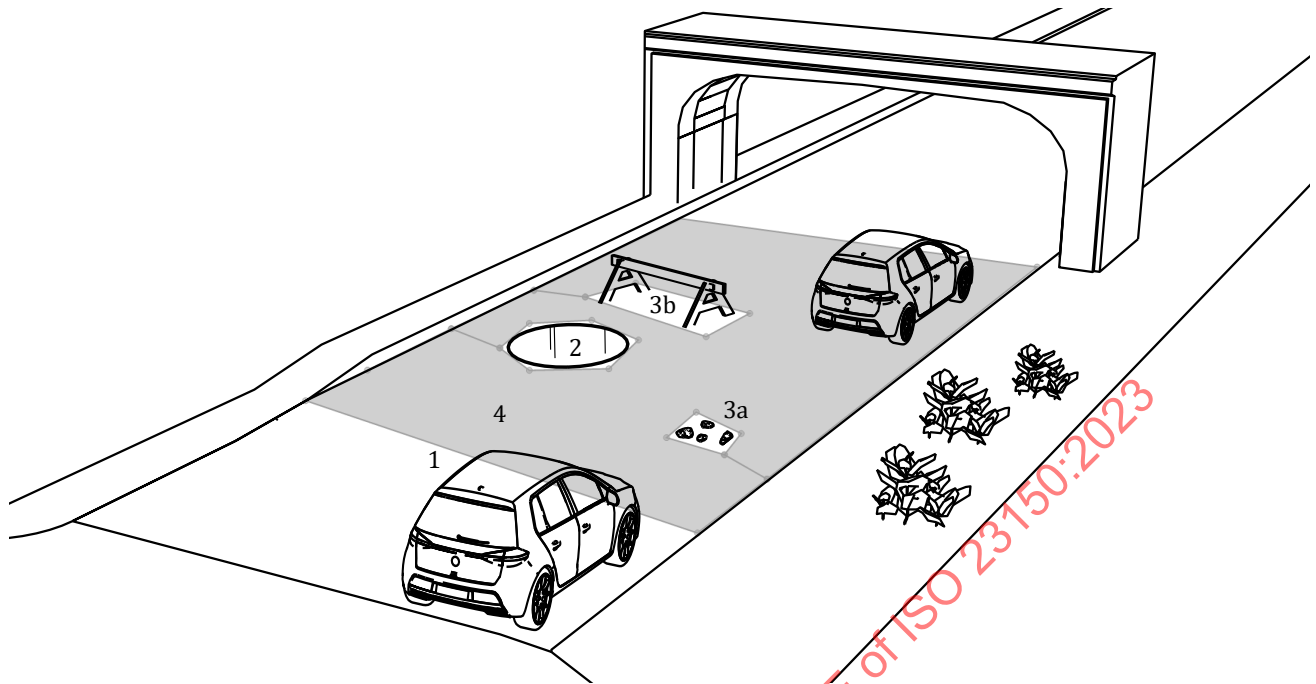
|  |   |           |
|--|---|-----------|
| FST_PerceivedLimitedByStatic           | The perceived measured free space area which is reduced by static objects. The free space area is not reduced by dynamic objects. The free space area is confirmed by active measurement of the sensor or the sensor cluster (see Figure A.24). | Mandatory |
| FST_PerceivedLimitedByStaticAndDynamic | The perceived free space area which is reduced by static and dynamic objects. The free space area is confirmed by active measurement of the sensor or the sensor cluster (see Figure A.25).   | Mandatory |



**Key**

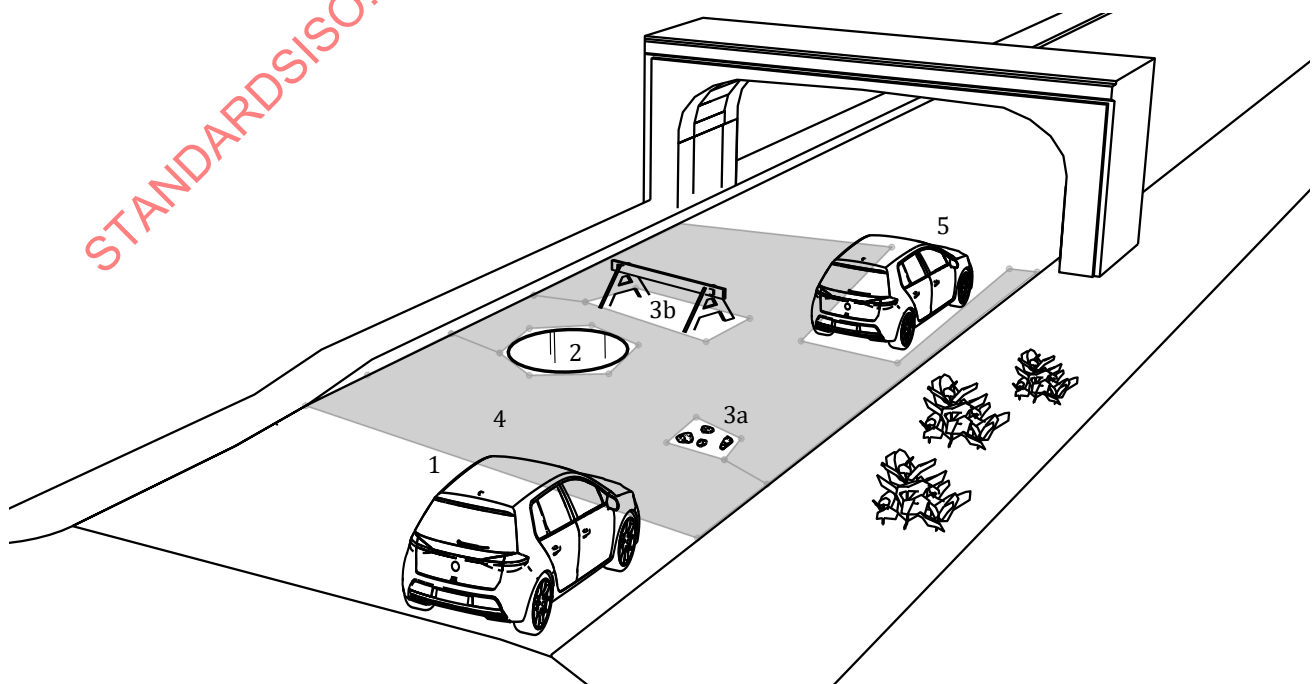
- 1 ego-vehicle
- 2 static object “LR\_Hole”
- 3a static object “LR\_Static” – small stones
- 3b static object “LR\_Static” – barrier
- 3c static object “LR\_Static” – vegetation
- 3d static object “LR\_Static” – wall
- ↕ height and extent of the limitation (static object)
- .....edge of the limitation (static object) – signal “Free space type” (A.2.201) := enumerator “FST\_LimitedByStatic”

**Figure A.23 — Example for signal “Free space type” (A.2.201) as enumerator “FST\_LimitedByStatic” and signal “Shape type – object level” (A.2.204) as enumerator “ST\_PolylineOpen”**

**Key**

- 1 ego-vehicle
- 2 static object "LR\_Hole"
- 3a static object "LR\_Static" – small stones
- 3b static object "LR\_Static" – barrier
- 4 free space area shape – signal "Free space type" (A.2.201) := enumerator "FST\_PerceivedLimitedByStatic"

**Figure A.24 — Example for signal "Free space type" (A.2.201) as enumerator "FST\_PerceivedLimitedByStatic" and signal "Shape type – object level" (A.2.204) as enumerator "ST\_PolylineClosed"**



**Key**

- 1 ego-vehicle
- 2 static object "LR\_Hole"
- 3a static object "LR\_Static" – small stones
- 3b static object "LR\_Static" – barrier
- 4 free space area shape – signal "Free space type" (A.2.201) := enumerator "FST\_PerceivedLimitedByStaticAndDynamic"
- 5 dynamic object "LR\_PotentiallyMovingObject" - vehicle

**Figure A.25 — Example for signal "Free space type" (A.2.201) as enumerator "FST\_PerceivedLimitedByStaticAndDynamic" and signal "Shape type – object level" (A.2.204) as enumerator "ST\_PolylineClosed"**

**A.2.202 Number of valid free space area shapes****Table A.320 — Signal: Number of valid free space area shapes**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid free space area shapes   |             |   |
| <b>Description</b> | The signal "Number of valid free space area shapes" (A.2.202) provides the current number of valid free space area shapes. |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

**A.2.203 Limitation geometry type****Table A.321 — Signal: Limitation geometry type**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Limitation geometry type  |             |   |
| <b>Description</b> | The signal "Limitation geometry type" (A.2.203) provides the information about the reference point where the geometric information (limitation and clearance) is located. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.322 — Enumeration: Limitation geometry type**

| <b>Name</b> | <b>Description</b>  | <b>RL enumerator</b> |
|-------------|---|----------------------|
| LGT_Vertex  | The limitation and clearance geometry information are assigned at the vertex point.   | Mandatory            |
| LGT_Edge    | The limitation and clearance geometry information are assigned to the property of the edge (for example, average, minimum or maximum value) between this and the next vertex point. Geometry information of the last vertex point may be invalid because it is not assigned to an edge. | Mandatory            |

**A.2.204 Shape type – object level****Table A.323 — Signal: Shape type – object level**

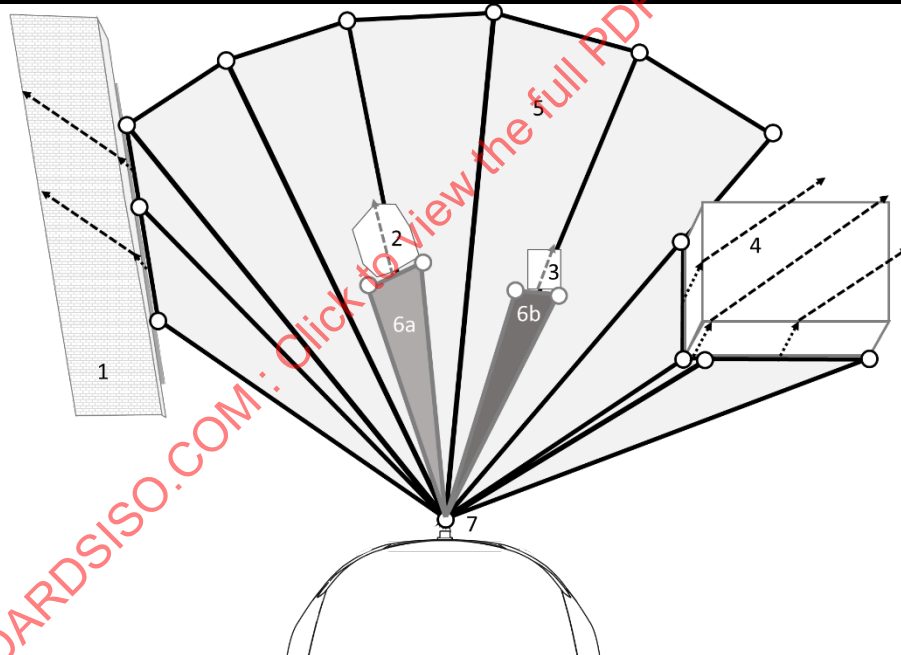
|                    |   |  |  |
|--------------------|---|--|--|
| <b>Name</b>        | Shape type – object level   |  |  |
| <b>Description</b> | <p>The signal "Shape type – object level" (A.2.204) provides the shape type which is used for the shape segment's borderline.</p> <p>Additional information:<br/>The shape defines how the vertex points are connected.</p> |  |  |



|                   |             |             |   |
|-------------------|-------------|-------------|---|
| <b>Value type</b> | enumeration | <b>Unit</b> | 1 |
|-------------------|-------------|-------------|---|

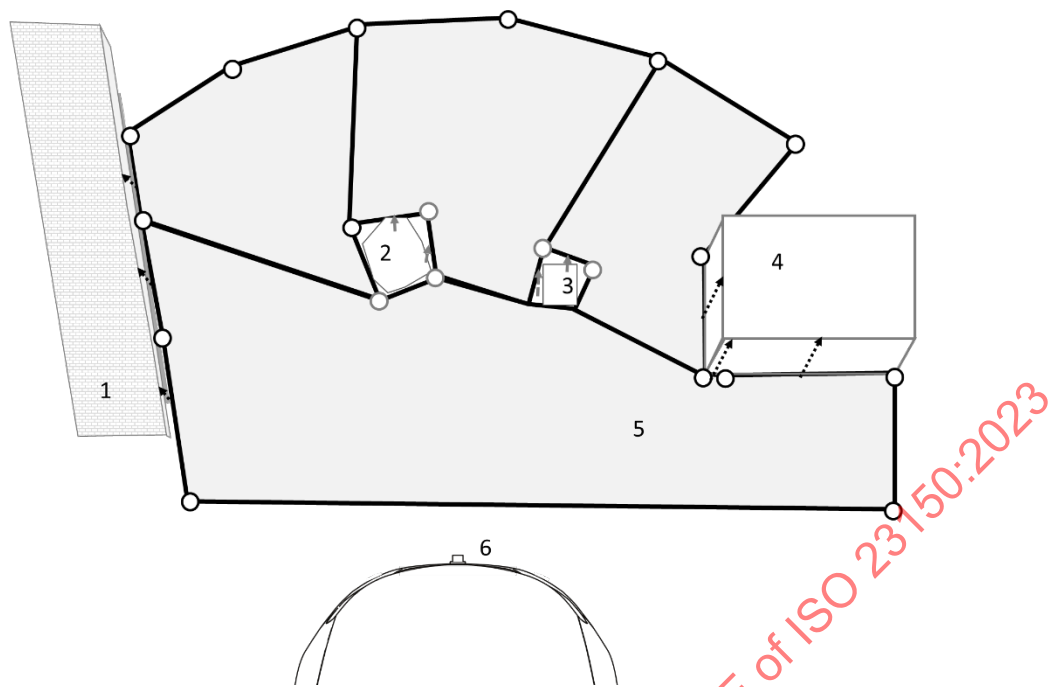
**Table A.324— Enumeration: Shape type – object level – Example enumerators**

| Name              | Description   | RL enumerator |
|-------------------|---|---------------|
| ST_Sector         | The shape is a triangular sector (2 or more explicit points and 1 implicit point). The area is defined by the two vertex points and a common vertex point of the ego-vehicle for all segments on road level [see signal “Sector common vertex point {x, y, z}” (A.2.205)].<br>The sectors may overlap. The limitation information of all free space sectors may overlap and may be combined to limit the free space area of a sector (see Figure A.26 – key 6). | “exemplary”   |
| ST_PolylineOpen   | The shape is an open polyline (2 or more points). The polyline is not closed with an edge between the last and first vertex point. A free space area is not perceived.  | “exemplary”   |
| ST_PolylineClosed | The shape is a closed polyline (3 or more points). The polyline is closed with an edge between the last and first vertex point (see Figure A.27 – key 5).<br>The closed polylines may not overlap.  | “exemplary”   |

**Key**

- 1 road edge
- 2 hole
- 3 static small object, for example, a box or a tire (that means the sensor can perceive beyond the object)
- 4 static large object (that means the sensor cannot perceive beyond the object)
- 5 free space area with multiple vertex points, which means with multiple segments (for example, the sensors maximum range)
- 6 free space area with 2 vertex points (for example, 6a a hole and 6b a box) – ego-vehicle reference vertex point
- 7 common vertex point - ego-vehicle with sensor [see signal “Sector common vertex point {x, y, z}” (A.2.205)]
- .....→ limitation {height}
- limitation {radial extent}

**Figure A.26 — Example for signal “Shape type – object level” (A.2.204) as enumerator “ST\_Sector” and signal “Limitation geometry type” (A.2.203) as enumerator “LGT\_Edge”**



Key

- 1 road edge
- 2 hole
- 3 static small object, for example, a box or a tire (that means the sensor can perceive beyond the object)
- 4 static large object (that means the sensor cannot perceive beyond the object)
- 5 free space area shape with multiple vertex points
- 6 ego-vehicle with sensor
- ..... limitation {height}

Figure A.27 — Example for signal “Shape type – object level” (A.2.204) as enumerator “ST\_PolylineClosed” and signal “Limitation geometry type” (A.2.203) as enumerator “LGT\_Edge”

A.2.205 Sector common vertex point {x, y, z}

Table A.325 — Signal: Sector common vertex point {x, y, z}

| Name        | Sector common vertex point {x, y, z}  |      |           |
|-------------|---|------|-----------|
| Description | The signal “Sector common vertex point {x, y, z}” (A.2.205) provides the measured longitudinal, lateral and vertical distance of the common vertex point for all sectors [see signal “Shape type – object level” (A.2.204) enumerator “ST_Sector”]. |      |           |
| Value type  | 3D vector real value  | Unit | (m, m, m) |

A.2.206 Sector common vertex point {x, y, z} – error

Table A.326 — Signal: Sector common vertex point {x, y, z} – error

| Name        | Sector common vertex point {x, y, z} – error  |      |  |
|-------------|---|------|--|
| Description | The signal “Sector common vertex point {x, y, z} – error” (A.2.206) provides the measured error of the signal “Sector common vertex point {x, y, z}” (A.2.205). |      |  |
| Value type  | scalar/vector/matrix real value<br>(see B.4.1)  | Unit | See signal “Sector common vertex point {x, y, z}” (A.2.205). |

**A.2.207 Sector common vertex point – confidence {x, y, z}****Table A.327 — Signal: Vertex point – confidence {x, y, z}**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Vertex point – confidence {x, y, z}  |             |   |
| <b>Description</b> | The signal “Sector common vertex point – confidence {x, y, z}” (A.2.207) provides the confidence for the signals of the sectors common vertex point [see signal “Sector common vertex point {x, y, z}” (A.2.205) and signal “Sector common vertex point {x, y, z} – error” (A.2.206)]. |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

**A.2.208 Limitation reason****Table A.328 — Signal: Limitation reason**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Limitation reason  |             |   |
| <b>Description</b> | The signal “Limitation reason” (A.2.208) describes the reason for the limit of the free space area at the given vertex point or edge [see signal “Limitation geometry type” (A.2.203)].<br><br>Additional information:<br>There is no definition for "overhang" of a vehicle load. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.329 — Enumeration: Limitation reason – Example enumerators**

| <b>Name</b>                | <b>Description</b>  | <b>RL enumerator</b> |
|----------------------------|---|----------------------|
| LR_None                    | The free space area is not limited at the current vertex point or edge. The vertex point or edge is at the connection of two shapes which describe a geometric complex free space area. | “exemplary”          |
| LR_Performance             | The free space area is limited at the current vertex point or edge by the maximum range or by performance reduction reasons.  | “exemplary”          |
| LR_Unclassified            | The free space area is limited at the current vertex point or edge by an unclassified object.   | “exemplary”          |
| LR_Static                  | The free space area is limited at the current vertex point or edge by a static object (SOI) or a road object (RDOI).  | “exemplary”          |
| LR_PotentiallyMovingObject | The free space area is limited at the current vertex point or edge by a potentially moving object (PMOI).   | “exemplary”          |
| LR_Hole                    | The free space area is limited at the current vertex point by a hole (that means a "negative limiting object").   | “exemplary”          |
| LR_SurfaceCondition        | The free space area is limited at the current vertex point or edge by almost flat media, for example, water, water puddle, oil spills, sand or ice.                                     | “exemplary”          |
| LR_OverheadStructure       | The free space area is limited by elevated objects (for example, lower bridges, tunnel or carpark entrance) (SOI).  | “exemplary”          |
| LR_ShadowEffect            | The free space area is limited by crosstalk or blooming effects.  | “exemplary”          |
| LR_Horizon                 | The free space area is limited by road topology.  | “exemplary”          |

A.2.209 Limitation moving probability

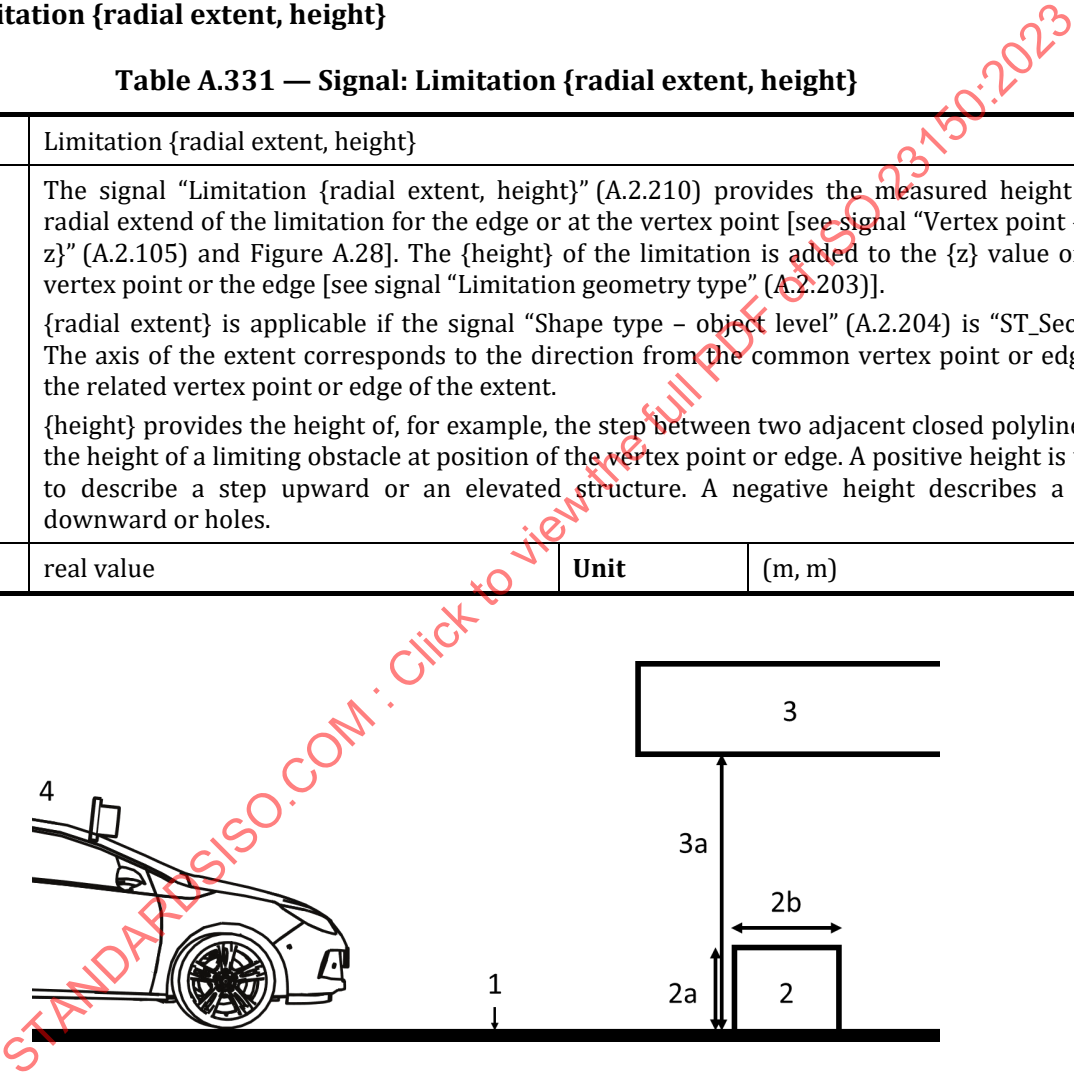
Table A.330 — Signal: Limitation moving probability

|             |  |      |   |
|-------------|--|------|---|
| Name        | Limitation moving probability  |      |   |
| Description | The signal “Limitation moving probability” (A.2.209) is only applicable to “FST_PerceivedLimitedByStaticAndDynamic” [see signal “Free space type” (A.2.201)] and may indicate a probability, whether the limitation is expected to be static or dynamic. |      |   |
| Value type  | [0...100] real value   | Unit | % |

A.2.210 Limitation {radial extent, height}

Table A.331 — Signal: Limitation {radial extent, height}

|             |   |      |        |
|-------------|---|------|--------|
| Name        | Limitation {radial extent, height}  |      |        |
| Description | <p>The signal “Limitation {radial extent, height}” (A.2.210) provides the measured height and radial extend of the limitation for the edge or at the vertex point [see signal “Vertex point {x, y, z}” (A.2.105) and Figure A.28]. The {height} of the limitation is added to the {z} value of the vertex point or the edge [see signal “Limitation geometry type” (A.2.203)].</p> <p>{radial extent} is applicable if the signal “Shape type – object level” (A.2.204) is “ST_Sector”. The axis of the extent corresponds to the direction from the common vertex point or edge to the related vertex point or edge of the extent.</p> <p>{height} provides the height of, for example, the step between two adjacent closed polylines or the height of a limiting obstacle at position of the vertex point or edge. A positive height is used to describe a step upward or an elevated structure. A negative height describes a step downward or holes.</p> |      |        |
| Value type  | real value  | Unit | (m, m) |



Key

- 1 road surface (free space area)
- 2 limitation – obstacle
- 2a {height} of the limitation “Limitation {radial extent, height}” (A.2.210)
- 2b {radial extent} of the limitation “Limitation {radial extent, height}” (A.2.210)
- 3 limitation – clearance and overhead object, for example, a bridge
- 3a {height} of the clearance “Clearance {height}” (A.2.213) for the limitation bridge
- 4 ego-vehicle with sensor

Figure A.28 — Example for signal “Limitation {radial extent, height}” (A.2.210) and signal “Clearance {height}” (A.2.213) of a free space area

**A.2.211 Limitation {radial extent, height} – error****Table A.332 — Signal: Limitation {radial extent, height} – error**

|                    |   |             |  |
|--------------------|---|-------------|--|
| <b>Name</b>        | Limitation {radial extent, height} – error  |             |  |
| <b>Description</b> | The signal “Limitation {radial extent, height} – error” (A.2.211) provides the error of the measured signal “Limitation {radial extent, height}” (A.2.210). |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal “Limitation {radial extent, height}” (A.2.210). |

**A.2.212 Limitation– confidence {radial extent, height}****Table A.333 — Signal: Limitation– confidence {radial extent, height}**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Limitation– confidence {radial extent, height}  |             |   |
| <b>Description</b> | The signal “Limitation– confidence {radial extent, height}” (A.2.212) provides the confidence of the signal “Limitation {radial extent, height}” (A.2.210) and the signal “Limitation {radial extent, height} – error” (A.2.211). |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

**A.2.213 Clearance {height}****Table A.334 — Signal: Clearance {height}**

|                    |  |             |        |
|--------------------|--|-------------|--------|
| <b>Name</b>        | Clearance {height}   |             |        |
| <b>Description</b> | <p>The signal “Clearance {height}” (A.2.213) provides the measured height of the limiting undriveable structure for the edge or at the vertex point [see signal “Vertex point {x, y, z}” (A.2.105)]. The {clearance} of the underdriveable structure is added to the {z} value of the vertex point or the edge [see signal “Limitation geometry type” (A.2.203) and Figure A.28].</p> <p>Additional information:<br/>The signal applies to the point of the vertex point or the edge depending on the signal “Limitation geometry type” (A.2.203).</p> |             |        |
| <b>Value type</b>  | real value   | <b>Unit</b> | (m, m) |

**A.2.214 Clearance {height} – error****Table A.335 — Signal: Clearance {height} – error**

|                    |  |             |  |
|--------------------|--|-------------|--|
| <b>Name</b>        | Clearance {height} – error   |             |  |
| <b>Description</b> | The signal “Clearance {height} – error” (A.2.214) provides the error of the measured clearance’s height. |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Clearance {height}” (A.2.213). |

**A.2.215 Clearance – confidence {height}****Table A.336 — Signal: Clearance – confidence {height}**

|             |                                 |  |  |
|-------------|---------------------------------|--|--|
| <b>Name</b> | Clearance – confidence {height} |  |  |
|-------------|---------------------------------|--|--|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | The signal “Clearance – confidence {height}” (A.2.215) provides the confidence of the clearance’s height [see signal “Clearance {height}” (A.2.213) and signal “Clearance {height} – error” (A.2.214)]. |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

### A.3 Feature level entity signals

Signals are defined for the entities of subclauses 8.3 and 8.4 in this subclause (see Table A.337-Table A.381). These interfaces are located at FLI. No FLI for radar or lidar sensors is defined.

The FLI originates from several sensors of a sensor cluster. Therefore, at feature level, the term sensor cluster is always used, even if a single sensor is serving the interface.

#### A.3.1 Existence probability – feature level

**Table A.337 — Signal: Existence probability – feature level**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Existence probability – feature level   |             |   |
| <b>Description</b> | <p>The signal “Existence probability – feature level” (A.3.1) provides the existence probability of the feature, based on history.</p> <p>Additional information:<br/>Use as a confidence measure where a low value means less confidence and a high value indicates strong confidence.</p> |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

#### A.3.2 Feature ID

**Table A.338 — Signal: Feature ID**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Feature ID   |             |   |
| <b>Description</b> | <p>The signal “Feature ID” (A.3.2) provides the unique identification number of the feature within the feature list.</p> <p>Additional information:<br/>The signal’s “Feature ID” (A.3.2) value is only unique for each sensor cluster’s feature entities of FLI within one measurement cycle. If interfaces use the signal “Feature ID” (A.3.2) to link, for example, detection entities with feature entities, the interfaces shall ensure the unambiguity of the referenced entities at each or multiple measurement, tracking or prediction cycle(s) [see, for example, cross interface optimisation “Source entities of a derived signal” (B.3.4)].</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

#### A.3.3 Object ID reference – feature level

**Table A.339 — Signal: Object ID reference – feature level**

|             |                                     |  |  |
|-------------|-------------------------------------|--|--|
| <b>Name</b> | Object ID reference – feature level |  |  |
|-------------|-------------------------------------|--|--|

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Description</b> | <p>The signal's "Object ID" (A.2.2) value defines the object to which this feature is associated to; invalid = "no associated object", for example, for the first feature cycle of the entity.</p> <p>Additional information:</p> <p>Refers to an object (for example, potentially moving object) with the highest probability for the association.</p> <p>The signal's "Object ID reference – feature level" (A.3.3) value could reference entities of the sensor cluster's OLI or entities which are used internally by the sensor cluster.</p> <p>Alternative A2I (B.3.2): the mapping of objects signal "Object ID" (A.2.2) and features signal "Object ID reference – feature level" (A.3.3) could be provided in a separate interface.</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

#### A.3.4 Time stamp difference – feature level

**Table A.340 — Signal: Time stamp difference – feature level**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Time stamp difference – feature level   |             |   |
| <b>Description</b> | <p>Each feature entity could provide an individual time stamp, specifically a time stamp difference to the signal's "Time stamp – prediction" (A.1.5.1) value of the interface.</p> <p>Additional information:</p> <p>The time stamp of the feature can be calculated as the signal "Time stamp – prediction" (A.1.5.1) of the FLI header for the cycle plus the signal "Time stamp difference – feature level" (A.3.4) of the feature.</p> |             |   |
| <b>Value type</b>  | real value  | <b>Unit</b> | s |

#### A.3.5 Number of valid observations – feature level

**Table A.341 — Signal: Number of valid observations – feature level**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Number of valid observations – feature level   |             |   |
| <b>Description</b> | <p>The signal "Number of valid observations – feature level" (A.3.5) provides the current number of valid tuples [specifically the signal "Time stamp reference – feature level" (A.3.6) and the signal "Observation status – feature level" (A.3.7)].</p> |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

#### A.3.6 Time stamp reference – feature level

**Table A.342 — Signal: Time stamp reference – feature level**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Time stamp reference – feature level   |             |   |
| <b>Description</b> | <p>The signal "Time stamp reference – feature level" (A.3.6) provides a reference to a previous sent interface message with the referenced time stamp for signal "Time stamp – &lt;...&gt;" (A.1.5).</p> |             |   |
| <b>Value type</b>  | real value   | <b>Unit</b> | s |

#### A.3.7 Observation status – feature level

**Table A.343 — Signal: Observation status – feature level**

|             |                                    |  |  |
|-------------|------------------------------------|--|--|
| <b>Name</b> | Observation status – feature level |  |  |
|-------------|------------------------------------|--|--|

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Description</b> | The signal “Observation status – feature level” (A.3.7) provides the observation status of the feature, which was recognised in a previous cycle [see signal “Time stamp reference – feature level” (A.3.6)]. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.344 — Enumeration: Observation status – feature level – Example enumerators**

| Name     | Description  | RL enumerator |
|----------|--|---------------|
| OS_True  | The feature was observed in the referenced cycle.  | “exemplary”   |
| OS_False | The feature was not observed in the referenced cycle. It may be predicted in the referenced cycle. | “exemplary”   |

**A.3.8 Feature grouping ID****Table A.345 — Signal: Feature grouping ID**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Feature grouping ID  |             |   |
| <b>Description</b> | <p>A sensor can group features from the FLI which are linked together. All features from the same entity have the same signal’s “Feature grouping ID” (A.3.8) value. The ID is unique considering all FLIs of a sensor cluster.</p> <p>EXAMPLE A pedestrian can be described with multiple camera shapes. All shapes of one pedestrian use the same “Feature grouping ID” (A.3.8) value.</p> <p>Additional information:<br/>The signal “Feature grouping ID” (A.3.8) := 0 if the feature is not associated with another feature.</p> |             |   |
| <b>Value type</b>  | integer value  | <b>Unit</b> | 1 |

**A.3.9 Number of valid shape classifications – feature level****Table A.346 — Signal: Number of valid shape classifications – feature level**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid shape classifications – feature level   |             |   |
| <b>Description</b> | The signal “Number of valid shape classifications – feature level” (A.3.9) provides the current number of valid tuples for shape classifications. The tuples are defined by the signals “Shape classification type – feature level” (A.3.10) and “Shape classification type – confidence – feature level” (A.3.11). |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.3.10 Shape classification type – feature level****Table A.347 — Signal: Shape classification type – feature level**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Shape classification type – feature level   |             |   |
| <b>Description</b> | The signal “Shape classification type – feature level” (A.3.10) provides the classification type for the shape. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |



**Table A.348 — Enumeration: Shape classification type – feature level – Example enumerators**

| Name                | Description  | RL enumerator |
|---------------------|--|---------------|
| SCT_Background      | The shape is classified as a background entity.            | “exemplary”   |
| SCT_Foreground      | The shape is classified as a foreground entity.            | “exemplary”   |
| SCT_Flat            | The shape is classified as a flat entity.                  | “exemplary”   |
| SCT_Upright         | The shape is classified as an upright entity.              | “exemplary”   |
| SCT_Ground          | The shape is classified as a ground entity.                | “exemplary”   |
| SCT_Building        | The shape is classified as a building entity.              | “exemplary”   |
| SCT_Vegetation      | The shape is classified as a vegetation entity.            | “exemplary”   |
| SCT_Road            | The shape is classified as a road entity.                  | “exemplary”   |
| SCT_NonRoad         | The shape is classified as a non-road entity.              | “exemplary”   |
| SCT_Pavement        | The shape is classified as a pavement entity.              | “exemplary”   |
| SCT_Pedestrian      | The shape is classified as a pedestrian entity.            | “exemplary”   |
| SCT_Vehicle         | The shape is classified as a vehicle entity.               | “exemplary”   |
| SCT_TrafficSign     | The shape is classified as a traffic sign entity.          | “exemplary”   |
| SCT_PedestrianFront | The shape is classified as a pedestrian front-view entity. | “exemplary”   |
| SCT_PedestrianSide  | The shape is classified as a pedestrian side-view entity.  | “exemplary”   |
| SCT_PedestrianRear  | The shape is classified as a pedestrian rear-view entity.  | “exemplary”   |

**A.3.11 Shape classification type – confidence – feature level****Table A.349 — Signal: Shape classification type – confidence – feature level**

| Name        | Shape classification type – confidence – feature level   |      |   |
|-------------|--|------|---|
| Description | The signal “Shape classification type – confidence – feature level” (A.3.11) provides the classification confidence of the shape segment’s possible content. |      |   |
| Value type  | [0...100] real value   | Unit | % |

**A.3.12 Colour value – feature level****Table A.350 — Signal: Colour value – feature level**

| Name        | Colour value – feature level  |      |   |
|-------------|---|------|---|
| Description | <p>The signal “Colour value – feature level” (A.3.12) provides the definition of the colour value(s). The number of values depends on the signal “Colour model type” (A.1.14).</p> <p>Additional information:<br/>For example, enumerator “CMT_RGB” has 3 colour values, specifically first value is red, second value is green, third value is blue.</p> |      |   |
| Value type  | [0...100] real value  | Unit | % |

## A.3.13 Colour tone – confidence – feature level

Table A.351 — Signal: Colour tone – confidence – feature level

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Colour tone – confidence – feature level   |             |   |
| <b>Description</b> | The signal “Colour tone – confidence – feature level” (A.3.13) provides the confidence of the shape area with the specified colour tone. The tone is defined by the signal “Colour model type” (A.1.14) and the list of signals “Colour value – feature level” (A.3.12). |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

## A.3.14 Shape type – feature level

Table A.352 — Signal: Shape type – feature level

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Shape type – feature level   |             |   |
| <b>Description</b> | <p>The signal “Shape type – feature level” (A.3.14) provides the shape type which is used for the shape segment’s borderline.</p> <p>Additional information:</p> <p>The shape defines how the referenced points are connected. The interpretation of the enumerators will be defined during the system design phase.</p> |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

Table A.353 — Enumeration: Shape type – feature level – Example enumerators

| <b>Name</b>   | <b>Description</b>                             | <b>RL enumerator</b> |
|---------------|--|----------------------|
| ST_Point      | The shape is a point.                          | “exemplary”          |
| ST_Box        | The shape is a box (2 or 3 points).            | “exemplary”          |
| ST_Ellipse    | The shape is an ellipse (2 or 3 points).       | “exemplary”          |
| ST_Polygon    | The shape is a polygon (3 or more points).     | “exemplary”          |
| ST_Polyline   | The shape is a polyline (2 or more points).    | “exemplary”          |
| ST_PointCloud | The shape is a point cloud (2 or more points). | “exemplary”          |

## A.3.15 Number of valid shape points – feature level

Table A.354 — Signal: Number of valid shape points – feature level

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid shape points – feature level  |             |   |
| <b>Description</b> | <p>The signal “Number of valid shape points – feature level” (A.3.15) provides the current number of valid shape points. The shape points are part of, for example, the polyline, which defines the border of the shape [see signal “Shape type – feature level” (A.3.14)].</p> <p>Additional information:</p> <p>The signal “Number of valid shape points – feature level” (A.3.15) <math>\geq 1</math>.</p> |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.3.16 Point existence probability – feature level****Table A.355 — Signal: Point existence probability – feature level**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Point existence probability – feature level  |             |   |
| <b>Description</b> | <p>The signal “Point existence probability – feature level” (A.3.16) provides the existence probability of the point [see signal “Position – feature level {x, y, z}” (A.3.17)]. Each point is only used by one shape [maybe one shape is sent in several entities with different shape classification confidences – see signal “Shape classification type – confidence – feature level” (A.3.11)].</p> <p>Additional information:<br/>Use as confidence measure where a low value means less confidence and a high value indicates strong confidence.</p> |             |   |
| <b>Value type</b>  | [0...100] real value   | <b>Unit</b> | % |

**A.3.17 Position – feature level {x, y, z}****Table A.356 — Signal: Position – feature level {x, y, z}**

|                    |  |             |           |
|--------------------|--|-------------|-----------|
| <b>Name</b>        | Position – feature level {x, y, z}   |             |           |
| <b>Description</b> | The signal “Position – feature level {x, y, z}” (A.3.17) provides the 3D point in ego-vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)]. |             |           |
| <b>Value type</b>  | 3D vector real value   | <b>Unit</b> | (m, m, m) |

**A.3.18 Position – feature level {x, y, z} – error****Table A.357 — Signal: Position – feature level {x, y, z} – error**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Position – feature level {x, y, z} – error  |             |   |
| <b>Description</b> | The signal “Position – feature level {x, y, z} – error” (A.3.18) provides the error representing the uncertainty of the state estimation of the signal “Position – feature level {x, y, z}” (A.3.17). |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal “Position – feature level {x, y, z}” (A.3.17). |

**A.3.19 Number of valid shape reference points – feature level****Table A.358 — Signal: Number of valid shape reference points – feature level**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid shape reference points – feature level  |             |   |
| <b>Description</b> | The signal “Number of valid shape reference points – feature level” (A.3.19) provides the current number of valid shape reference points. Each signal’s “Position – feature level {x, y, z}” (A.3.17) position of a shape reference point is part of the shape surface. |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

**A.3.20 Shape surface normal {x, y, z}****Table A.359 — Signal: Shape surface normal {x, y, z}**

|             |                                |  |  |
|-------------|--------------------------------|--|--|
| <b>Name</b> | Shape surface normal {x, y, z} |  |  |
|-------------|--------------------------------|--|--|

|                    |   |             |           |
|--------------------|---|-------------|-----------|
| <b>Description</b> | The signal “Shape surface normal {x, y, z}” (A.3.20) provides the normal vector of the shape’s approximated tangential plane at the shape reference point [see signal “Position – feature level {x, y, z}” (A.3.17)].<br><br>Additional information:<br>The vector is normalised. |             |           |
| <b>Value type</b>  | 3D vector real value  | <b>Unit</b> | (1, 1, 1) |

### A.3.21 Shape surface normal {x, y, z} – error

**Table A.360 — Signal: Shape surface normal {x, y, z} – error**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Shape surface normal {x, y, z} – error  |             |   |
| <b>Description</b> | The signal “Shape surface normal {x, y, z} – error” (A.3.21) provides the error of the signal “Shape surface normal {x, y, z}” (A.3.20) which is the normal of the shape’s approximated tangential plane at the shape reference point [see corresponding signal “Position – feature level {x, y, z}” (A.3.17)]. |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal “Shape surface normal {x, y, z}” (A.3.20). |

### A.3.22 Translation rate {x, y, z} – feature level

**Table A.361 — Signal: Translation rate {x, y, z} – feature level**

|                    |   |             |                 |
|--------------------|---|-------------|-----------------|
| <b>Name</b>        | Translation rate {x, y, z} – feature level  |             |                 |
| <b>Description</b> | The signal “Translation rate {x, y, z} – feature level” (A.3.22) provides the scaled translation of the shape’s approximated tangential plane at the shape reference point [see corresponding signal “Position – feature level {x, y, z}” (A.3.17)]. Longitudinally (along the view axis) that is the inverse time to collision.<br>No motion → (0, 0, 0) m/s |             |                 |
| <b>Value type</b>  | 3D vector real value  | <b>Unit</b> | (m/s, m/s, m/s) |

### A.3.23 Translation rate {x, y, z} – feature level – error

**Table A.362 — Signal: Translation rate {x, y, z} – feature level – error**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Translation rate {x, y, z} – feature level – error  |             |   |
| <b>Description</b> | The signal “Translation rate {x, y, z} – feature level – error” (A.3.23) provides the error of the signal “Translation rate {x, y, z} – feature level” (A.3.22) of the shape’s approximated tangential plane at the shape reference point [see corresponding signal “Position – feature level {x, y, z}” (A.3.17)]. |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal “Translation rate {x, y, z} – feature level” (A.3.22). |

### A.3.24 Rotation rate {yaw, pitch, roll} – feature level

**Table A.363 — Signal: Rotation rate {yaw, pitch, roll} – feature level**

|             |  |  |  |
|-------------|--|--|--|
| <b>Name</b> | Rotation rate {yaw, pitch, roll} – feature level |  |  |
|-------------|--|--|--|

|                    |   |             |                       |
|--------------------|---|-------------|-----------------------|
| <b>Description</b> | The signal “Rotation rate {yaw, pitch, roll} – feature level” (A.3.24) provides the rotation change of the shape’s approximated tangential plane at the shape reference point [see corresponding signal “Position – feature level {x, y, z}” (A.3.17)].<br>No rotation change $\rightarrow (0, 0, 0)$ rad/s |             |                       |
| <b>Value type</b>  | 3D vector real value  | <b>Unit</b> | (rad/s, rad/s, rad/s) |

### A.3.25 Rotation rate {yaw, pitch, roll} – error – feature level

**Table A.364 — Signal: Rotation rate {yaw, pitch, roll} – error – feature level**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Rotation rate {yaw, pitch, roll} – error – feature level   |             |   |
| <b>Description</b> | The signal “Rotation rate {yaw, pitch, roll} – error – feature level” (A.3.25) provides the error of the signal’s “Rotation rate {yaw, pitch, roll} – feature level” (A.3.24) change of the shape’s approximated tangential plane at the shape reference point [see corresponding signal “Position – feature level {x, y, z}” (A.3.17)]. |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Rotation rate {yaw, pitch, roll} – feature level” (A.3.24). |

### A.3.26 Scale change – feature level

**Table A.365 — Signal: Scale change – feature level**

|                    |  |             |     |
|--------------------|--|-------------|-----|
| <b>Name</b>        | Scale change – feature level   |             |     |
| <b>Description</b> | The signal “Scale change – feature level” (A.3.26) provides the mean scale ratio of the shape’s approximated tangential plane at the shape reference point [see corresponding signal “Position – feature level {x, y, z}” (A.3.17)].<br>No change in size $\Rightarrow$ ratio of 100 %.<br><br>Additional information:<br>Describes the lateral shift and size change of the shape segment over time.<br>The signal “Scale change – feature level” (A.3.26) is normalised. |             |     |
| <b>Value type</b>  | real value   | <b>Unit</b> | %/s |

### A.3.27 Scale change – feature level – error

**Table A.366 — Signal: Scale change – feature level – error**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Scale change – feature level – error  |             |   |
| <b>Description</b> | The signal “Scale change – feature level – error” (A.3.27) provides the error of the signal “Scale change – feature level” (A.3.26) of the shape’s approximated tangential plane at the shape reference point [see corresponding signal “Position – feature level {x, y, z}” (A.3.17)]. |             |   |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal “Scale change – feature level” (A.3.26). |

### A.3.28 Number of valid ultrasonic feature classifications

**Table A.367 — Signal: Number of valid ultrasonic feature classifications**

|             |  |  |  |
|-------------|--|--|--|
| <b>Name</b> | Number of valid ultrasonic feature classifications |  |  |
|-------------|--|--|--|

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Description</b> | The signal “Number of valid ultrasonic feature classifications” (A.3.28) provides the current number of valid tuples for ultrasonic feature classifications. The tuples are defined by the signals “Ultrasonic feature classification type” (A.3.29) and “Ultrasonic feature classification type – confidence” (A.3.30). |             |   |
| <b>Value type</b>  | [0...] integer value   | <b>Unit</b> | 1 |

### A.3.29 Ultrasonic feature classification type

**Table A.368 — Signal: Ultrasonic feature classification type**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Ultrasonic feature classification type   |             |   |
| <b>Description</b> | The signal “Ultrasonic feature classification type” (A.3.29) contains information about the current measurement of this feature. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.369 — Enumeration: Ultrasonic feature classification type – Example enumerators**

| <b>Name</b>      | <b>Description</b>  | <b>RL enumerator</b> |
|------------------|---|----------------------|
| UFCT_Point       | The feature is defined by one point and optional by a height value.         | “exemplary”          |
| UFCT_LineSegment | The feature is defined by two or more points and optional by height values. | “exemplary”          |

### A.3.30 Ultrasonic feature classification type – confidence

**Table A.370 — Signal: Ultrasonic feature classification type – confidence**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Ultrasonic feature classification type – confidence   |             |   |
| <b>Description</b> | The signal “Ultrasonic feature classification type – confidence” (A.3.30) provides the confidence of the ultrasonic feature type. |             |   |
| <b>Value type</b>  | [0...100] real value  | <b>Unit</b> | % |

### A.3.31 Number of valid points

**Table A.371 — Signal: Number of valid points**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Number of valid points  |             |   |
| <b>Description</b> | The signal “Number of valid points” (A.3.31) provides the current number of valid points. |             |   |
| <b>Value type</b>  | [0...] integer value  | <b>Unit</b> | 1 |

### A.3.32 Orientation – feature level {pitch}

**Table A.372 — Signal: Orientation – feature level {pitch}**

|                    |  |             |     |
|--------------------|--|-------------|-----|
| <b>Name</b>        | Orientation – feature level {pitch}  |             |     |
| <b>Description</b> | The signal “Orientation – feature level {pitch}” (A.3.32) provides the orientation of the feature. |             |     |
| <b>Value type</b>  | 1D vector real value   | <b>Unit</b> | rad |

**A.3.33 Orientation – feature level {pitch} – error****Table A.373 — Signal: Orientation – feature level {pitch} – error**

|                    |  |             |  |
|--------------------|--|-------------|--|
| <b>Name</b>        | Orientation – feature level {pitch} – error  |             |  |
| <b>Description</b> | The signal “Orientation – feature level {pitch} – error” (A.3.33) provides the error value of the signal “Orientation – feature level {pitch}” (A.3.32). |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Orientation – feature level {pitch}” (A.3.32). |

**A.3.34 Extent {height} – feature level****Table A.374 — Signal: Extent {height} – feature level**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Extent {height} – feature level  |             |   |
| <b>Description</b> | <p>The signal “Extent {height} – feature level” (A.3.34) provides the height of the point [see signal “Position – feature level {x, y, z}” (A.3.17)]. That means, the upper edge of the feature is specified.</p> <p>Additional information:<br/>The signal “Extent {height} – feature level” (A.3.34) is the Z-axis offset of the signal “Position – feature level {x, y, z}” (A.3.17). If {z} is not specified, the height is with respect to the origin of the ego-vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)].</p> |             |   |
| <b>Value type</b>  | real value   | <b>Unit</b> | m |

**A.3.35 Extent {height} – feature level – error****Table A.375 — Signal: Extent {height} – feature level – error**

|                    |  |             |  |
|--------------------|--|-------------|--|
| <b>Name</b>        | Extent {height} – feature level – error  |             |  |
| <b>Description</b> | The signal “Extent {height} – feature level – error” (A.3.35) provides the error value of the signal “Extent {height} – feature level” (A.3.34). |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)   | <b>Unit</b> | See signal “Extent {height} – feature level” (A.3.34). |

**A.3.36 Velocity {x, y} – feature level****Table A.376 — Signal: Velocity {x, y} – feature level**

|                    |   |             |            |
|--------------------|---|-------------|------------|
| <b>Name</b>        | Velocity {x, y} – feature level   |             |            |
| <b>Description</b> | The signal “Velocity {x, y} – feature level” (A.3.36) provides the velocity over ground of the feature which is specified in the ego-vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)]. |             |            |
| <b>Value type</b>  | 2D vector real value  | <b>Unit</b> | (m/s, m/s) |

**A.3.37 Velocity {x, y} – feature level – error****Table A.377 — Signal: Velocity {x, y} – feature level – error**

|             |   |  |  |
|-------------|---|--|--|
| <b>Name</b> | Velocity {x, y} – feature level – error |  |  |
|-------------|---|--|--|

|                    |   |             |  |
|--------------------|---|-------------|--|
| <b>Description</b> | The signal “Velocity {x, y} – feature level – error” (A.3.37) provides the error values for the signal “Velocity {x, y} – feature level” (A.3.36) in the ego-vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)]. |             |  |
| <b>Value type</b>  | scalar/vector/matrix real value<br>(see B.4.1)  | <b>Unit</b> | See signal “Velocity {x, y} – feature level” (A.3.36). |

### A.3.38 Trilateration status

**Table A.378 — Signal: Trilateration status**

|                    |   |             |   |
|--------------------|---|-------------|---|
| <b>Name</b>        | Trilateration status  |             |   |
| <b>Description</b> | The signal “Trilateration status” (A.3.38) provides the information whether the feature is trilaterated with multiple signal ways or is not trilaterated. |             |   |
| <b>Value type</b>  | enumeration   | <b>Unit</b> | 1 |

**Table A.379 — Enumeration: Trilateration status – Example enumerators**

| <b>Name</b>        | <b>Description</b>  | <b>RL enumerator</b> |
|--------------------|---|----------------------|
| TS_Trilaterated    | The measurement of the signal’s “Position – feature level {x, y, z}” (A.3.17) point is based on at least three detections for a 2D information. | “exemplary”          |
| TS_NotTrilaterated | The measurement of the signal’s “Position – feature level {x, y, z}” (A.3.17) point is based on less than three detections.                     | “exemplary”          |

### A.3.39 Measurement status – feature level

**Table A.380 — Signal: Measurement status – feature level**

|                    |  |             |   |
|--------------------|--|-------------|---|
| <b>Name</b>        | Measurement status – feature level   |             |   |
| <b>Description</b> | The signal “Measurement status – feature level” (A.3.39) provides the information about the measurement status of the feature. |             |   |
| <b>Value type</b>  | enumeration  | <b>Unit</b> | 1 |

**Table A.381 — Enumeration: Measurement status – feature level – Example enumerators**

| <b>Name</b>       | <b>Description</b>                                | <b>RL enumerator</b> |
|-------------------|---|----------------------|
| MS_Initialisation | No information is available.                      | “exemplary”          |
| MS_Tracked        | It was not measured in the current cycle.         | “exemplary”          |
| MS_Measured       | Current position of this feature was measured.    | “exemplary”          |
| MS_Delete         | Tracking will be deleted in the next cycle.       | “exemplary”          |
| MS_New            | It is the first cycle of tracking of the feature. | “exemplary”          |

## A.4 Detection level entity signals

Signals are defined for the entities of subclauses 9.3-9.6 in this subclause (see Table A.382-Table A.429). These interfaces are located at the DLI. Normally the DLI originates from one sensing element. Therefore, at detection level, the term sensor is always used, even if a combination of emitting as well as sensing elements are serving the interface.

### A.4.1 Existence probability – detection level