OCORA

Open CCS On-board Reference Architecture

Application Layer Interface Specification ATO/ CCS-TCMS Interface – ATO Functionality

Document ID: OCORA-TWS15-030

Version: 2.00

Release: Delta

Date: 30.06.2021

Revision history

Version	Change Description	Initial	Date of change
2.00	Official version for OCORA Delta Release	BJ	30.06.2021

Table of contents

Table of figures

Table of tables

References

Reader's note: please be aware that the numbers in square brackets, e.g. [1], as per the list of referenced documents below, is used throughout this document to indicate the references to external documents. Wherever a reference to a TSI-CCS SUBSET is used, the SUBSET is referenced directly (e.g. SUBSET-026). OCORA always reference to the latest available official version of the SUBSET, unless indicated differently.

[1] OCORA-BWS01-010 – Release Notes

[2] OCORA-BWS01-020 – Glossary

[3] OCORA-BWS01-030 – Question and Answers

[4] OCORA-BWS01-040 – Feedback Form

[5] OCORA-BWS03-010 – Introduction to OCORA

[6] OCORA-BWS04-010 – Problem Statements

[7] OCORA-BWS08-010 – Methodology

[8] OCORA-BWS08-020 – Tooling

[9] OCORA-TWS01-030 – System Architecture

[10] OCORA-TWS04-010 – Functional Vehicle Adapter – Introduction

[11] OCORA-TWS04-011 – Functional Vehicle Adapter – Requirements

[12] OCORA-TWS04-012 – Functional Vehicle Adapter – Standard Communication Interface Specification

[13] OCORA-TWS04-013 – Functional Vehicle Adapter – Design Guideline

[14] OCORA-TWS15-010 – Prototyping – ATO Demonstrator - Case Study - S2R-IP-5-ARCC

[15] OCORA-TWS15-020 – CCS-TCMS-Interface-ETCS-Functionality

[16] ATO over ERTMS System Requirement Specification – SUBSET-125 – UNISIG – Version 0.1.0

[17] ATO-OB / TCMS Interface Specification FIS + FFFIS – SUBSET-139 – UNISIG – Version 0.0.14

[18] ATO over ETCS: ATO-OB / ETCS-OB FFFIS Application Layer – SUBSET-130 – UNISIG – Version 010-W3

[19] Train Interface - FIS – SUBSET-34 – UNISIG – Version 3.1.0

[20] Information transmission in the train (train bus) - General dispositions – UIC 556 – UIC – Version 4

[21] Glossary of Terms and Abbreviations – SUBSET-023 – UNISIG – Version 3.1.0

[22] TSI LOC & PAS – 2011/291/EU & 2012/464/EU – European Commission

[23] ATO OVER ETCS GLOSSARY – EUG 13E154– EUG

[24] ETCS System Requirements Specification – SUBSET-026 – UNISIG – Version 3.6.0

[25] SAE Truck and Bus Control Communications Network – SAE J1393 standard – Society of Automotive Engineers

[26] ATO over ETCS, System Interface Specification, Communication Layers – SUBSET-143 – UNISIG – Version 0.0.5

[27] Train Interface FFFIS – SUBSET-119 – UNISIG – Version 1.0.15

Abbreviations and definitions

- For ATO abbreviations and definitions see SUBSET-125.[16]
- For ETCS abbreviations and definitions see SUBSET-023 [21].
- Other definitions used in this document:

Term	***Explanation***
Dynamic brake	The brake system that realizes the braking effort using the propulsion (traction converter, hydrodynamic converter). From the principle, the braking effort can be realized only if the speed of the vehicle is greater than specific value.
EMU (for purpose of this document)	Electric multiple unit, the vehicle with common brake control (separate control of dynamic and train air brake is not possible). DMUs, railbuses or electric rail cars are also included in this term.
Functional Vehicle Adaptor	The Functional Vehicle Adapter (FVA) is a piece of software deployed on the OCORA Computing Platform, or on the OCORA Gateway, or on the TCMS. Its job is to provide an OCORA unified and standardized interface towards the CCS applications and services for vehicle functions and vehicle information needed by the OCORA on-board applications and services. Although the TSI-CCS subsets 034, 119, and 139 are defining the interface to the TCMS system, vehicle from different suppliers and especially from different generations have still different interfaces implemented. This adapter allows to map, on a functional level, the commands sent, and the information received from a specific TCMS into the OCORA standard. In addition, the FVA can also be used to integrate vehicles without a TCMS
Future Railway Mobile Communication System	The Future Railway Mobile Communication System (FRMCS) is the future worldwide telecommunication system designed by UIC, in close cooperation with the different stakeholders from the rail sector, as the successor of GSM-R but also as a key enabler for rail transport digitalisation.
Locomotive, Loco	the traction vehicle with independently controlled dynamic and train air brakes.
Mandatory Data	Vehicle data that are part of the minimal set of data required for safe and TSI conformal TCMS operation
Mandatory Functions	Vehicle functions that are part of the minimal set of functions required for safe and TSI conformal TCMS operation
Specific Vehicle Interface	Functional Module that ensures data exchange with the vehicle for data that can’t be handled by the Subset-139 FFFIS [17] and/ or the TCMS
Train Control & Management System	Train Control & Management System (TCMS) is a train-borne distributed control system. It comprises computer devices and software, human-machine interfaces, digital and analogue input/ output (I/O) capability and the data networks to connect all these together in a secure and fault-resistant manner. Train Control & Management System (TCMS) is a train-borne distributed control and command system.

- Other abbreviations used in this document:

Term	***Explanation***
Bitset	A set of binary signals that are transmitted together
Bool	Boolean (binary) signal
AV	See OCORA AV
CCS	Command and Control System
Dir	Direction
Enum	Enumerated (limited) set of values
EXT	FVA Interface for data exchange with external functions
FRMCS	Future Railway Mobile Communication System

Term	***Explanation***
FVA	Functional Vehicle Adaptor
Num	Numeric signal (continuous value)
OCORA AV	Automated Vehicle System
SVI	Specific Vehicle Interface
TCMS	Train Control & Management System

1 Introduction

1.1 Purpose of the document

The purpose of this document is to publish the prototype model developed by Deutsche Bahn AG. The prototype was implemented to foster and verify a transparent functional interface between the CCS on-board and the physical train unit (TCMS) for ATO Vehicle up to GoA2 functionality.

This document is addressed to experts in the CCS domain and to any other person, interested in the OCORA concepts for on-board CCS. The reader is invited to provide feedback to the OCORA collaboration and can, therefore, engage in shaping OCORA. Feedback to this document and to any other OCORA documentation can be given by using the feedback form [4].

If you are a railway undertaking, you may find useful information to compile tenders for OCORA compliant CCS building blocks, for tendering complete on-board CCS system, or also for on-board CCS replacements for functional upgrades or for life-cycle reasons.

If you are an organization interested in developing on-board CCS building blocks according to the OCORA standard, information provided in this document can be used as input for your development.

1.2 Applicability of the document

The document is currently considered informative and the findings are being integrated progressively in other OCORA documentation (e.g. in the “Functional Vehicle Adapter - Design Guideline” [13]). This document will be removed from the OCORA publications, once this process is completed,

1.3 Context of the document

This document is published as part of the OCORA Delta release, together with the documents listed in the release notes [1]. Before reading this document, it is recommended to read the Release Notes [1]. If you are interested in the context and the motivation that drives OCORA we recommend to read the Introduction to OCORA [5], and the Problem Statements [6]. The reader should also be aware of the Glossary [2] and the Question and Answers [3].

Furthermore, this document must be seen in the technical context of the Functional Vehicle Adapter. Therefore we invite to read the Functional Vehicle Adapter introduction document [10] which provides an overview of the concept, in [11] the Functional Vehicle Adapter requirements are defined while in [12] the interface to the different CCS on-board applications is developed.

1.4 Disclaimer

- This specification and referred models are following state of the art engineering rules, best practice and proofed implementation work.
- Some possible improvements are already identified and the specification and models will be optimized and maintained by feedback from experts, implementation and application within the future release phases and process defined by OCORA as published on https://github.com/OCORA-Public/Publication.
- The technical solutions developed by OCORA must not favour any particular product or supplier. Technical solutions shall allow a variety of products and methods/process.

2 Integration specification for an ATO on-board system

2.1 Challenge and solution

2.1.1.1 The ATO- TCMS interface is subject to a standardization effort for new vehicles, resulting in the definition of Subset-139 [17] / 143 [26].

2.1.1.2 A large part of ATO deployment will however affect existing vehicles with various TCMS concepts and -architectures. Not all the aspects of ATO integration on legacy vehicles are covered by Subset-139 [17] / 143 [26]. For more info please refer to Appendix 7.

2.1.1.3 In the context of the OCORA effort, the current version of this document is a first iteration. It is following the concept of the OCORA FVA (see [10], Functional Vehicle Adapter - Introduction) as also described in the figure below. This document describes already the logical concept of the ATO- specific part of the FVA, while retaining the data formats and -coding of the existing Onboard Units and the existing TCMS. During the following iterations, the abstraction on the application layer will be developed further, so that at the end the OCORA AV system needs no prior knowledge about the vehicle, all vehicle specifics are handled in the FVA or in the related systems that are described in this specification. In the current state of the design, which is aimed at the TSI 2022 baseline, this abstraction is already partially realized.

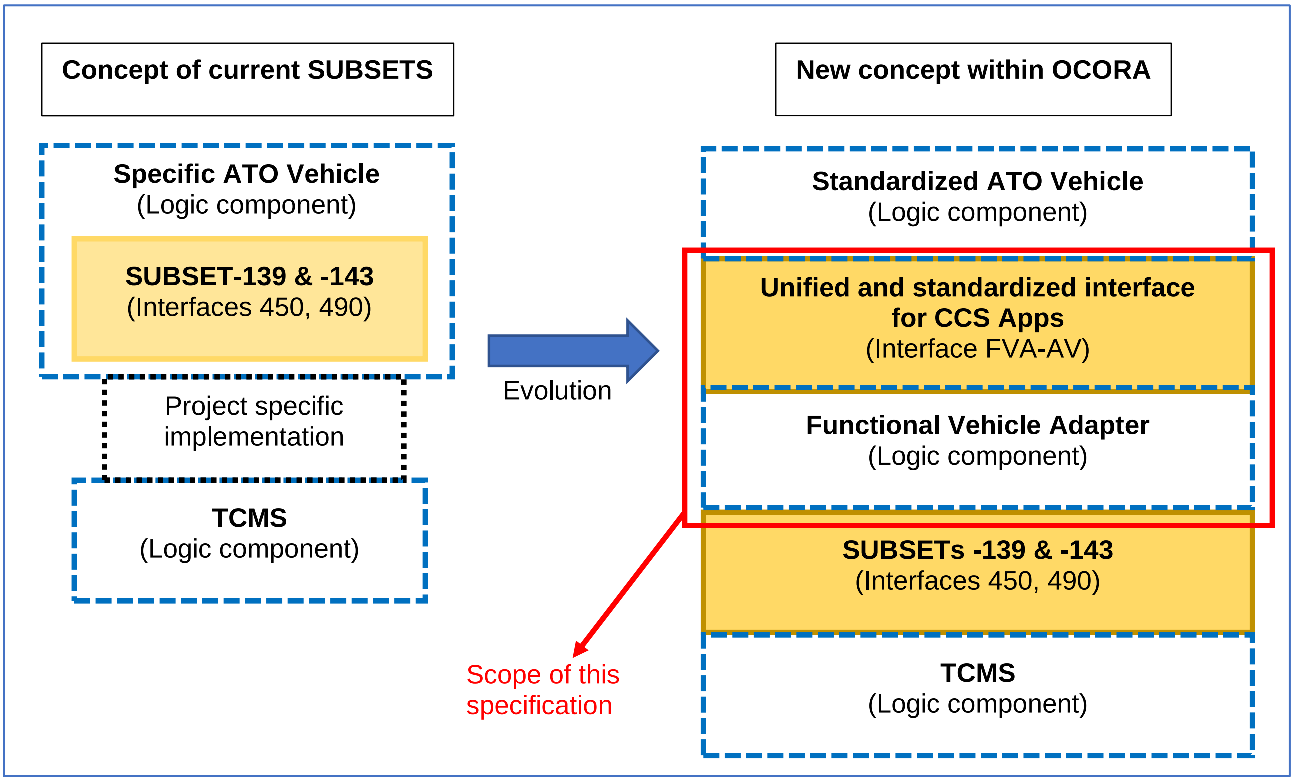


Figure 1: Scope of this document and relationship to OCORA architecture

2.1.1.4 Subset-139 [17] / 143 [26] is leaving a lot of freedom to the designer of the OBU/ TCMS interface. This document is intended to give more precise guidance on various TCMS configurations and – situations.

2.1.1.5 This specification is to be understood in context of Subset-139 [17]. It describes the following:

- standardized application interface for the ATO core application
- additional data TCMS interface for data that is outside of Subset-139 [17] / 143 [26]
- Functional Vehicle Adapter including its logic.

2.1.1.6 The goal is to enable a standardized ATO core application that can be used without modification on any vehicle. The specific vehicle adaptations shall be implemented in the Functional Vehicle Adapter specifically for each vehicle type/class. This specification is to be understood in the context of Subset-139 [17]. It describes an application- layer interface which uses Subset-139 [17] / 143 [26] for communication with the TCMS.

2.1.1.7 This document describes the ATO perspective on the TCMS interface.

2.1.1.8 This document describes a standardized interface for the ATO Onboard Unit. The ATO can access vehicle functions and vehicle data provided by the TCMS and/ or by Specific Vehicle Interface adapted to a particular project.

2.1.1.9 The interface is intended to be stable and independent of the architecture and features of the vehicle. However, a minimum set of functions and data needs to be provided by the combination of the TCMS and a Specific Vehicle Interface in order to enable the vehicle for ATO functionality. These functions and data are marked as mandatory in this document.

2.1.1.10 The solution described in this document provides standardised interface to the S2R Subset-139 application layer [17]. Any project- specific configurations and settings are encapsulated by the Functional Vehicle Adaptor.

2.2 Compliance with the TSI/UNISIG/S2R documents and Configuration Management

2.2.1.1 This document is compliant with existing TSI/UNISIG/S2R subsets and updated once the subsets is changed. Feedback from the implementations will result in updating and enriching this document. In a future step, OCORA will propose a detailed configuration management to support technical compatibility.

2.2.1.2 Compliance with subset 139 [17]: This work is mainly based and fully compliant on the Subset 139 Train Interface FFFIS [17], which remains untouched and stable. However, this work will give an overview about possible steps and necessary information towards a full FFFIS Plug & Play solution.

2.2.1.3 Compliance with subset 143 [26]: the subset 143 [26] describes the communication layer for ATO system as specified within the subset 139 [17] / 125 [16]. This specification is fully compatible to the subset 143 [26] layer, since the scope of this work is to describe the ATO application layer, which must be fully independent from a communication layer following the OSI IEC 61375-3-4:2014 according to the OCORA requirements.

2.3 Relation and reference of Model Based System Engineering to this work

2.3.1.1 OCORA has chosen a model-based system engineering methodology to ensure the quality, completeness, maintainability and evolvability of OCORA specifications. This document has to be understood as the outcome of such modelling process. For the modelling process, OCORA makes use of the Scade tool and the “Lustre” formal language to describe the formal model. The Scade Code Generator is CENELEC 50128 certified.
The formal model will be used for generating complete test cases to support the modular safety concept and test the correctness of the implementation. This modular safety concept will be developed in the next OCORA releases to support the certification and V&V process.

2.4 Implementation procedure

This specification is based on a systematic analysis of the following documents and standards:

- Subset-026 System Requirements Specification [24]
- Subset-139 Train Interface FFFIS [17]
- Subset-130 ATO-OB / ETCS-OB FFFIS Application Layer [18]
- Subset-125 ATO over ERTMS System Requirement Specification [16]
- Subset-143 ATO over ETCS. System Interface Specification Communication Layers [26]

The analysis has been carried out as follows:

- All the documents were transferred to a documentation/ requirements management system
- A requirements traceability matrix has been derived
- The gaps have been analysed:
  - The main parameters of the analysis were consistency of data flows and a functional analysis based on S2R specification and testing input.
- A formal model has been developed. The scope of the formal model encompasses:
  - The API Exposed to the ATO
  - The Functional Vehicle Adaptor
  - The Application Layer of the Subset-139 FFFIS
  - The Specific Vehicle Interface

with respect to best proofed implementations, state of the art engineering experience and best practices.

The formal model enables static analysis for data coupling, consistency, completeness and determinism of the complete data flow between the ATO Onboard and the vehicle. It is also possible to use the model as an executable specification and as a formal basis for the validation of project- specific implementations.

2.5 Document structure

This document is structured as follows:

After a general introduction (this section), chapter 4 Architecture, provides a discussion of the general architecture of the ATO/Train Interface. We also discuss several reference solutions for different types of vehicle in order to highlight the modular and layered approach.

Chapter 5: ATO Core Interface, describes the data interface provided to the ATO core, introduction all variables and packets that are specified.

Chapter 6: ATO Function Vehicle Adapter describes the functionalities that allow the adaption of the ATO/ Train interface to various TCMS and Vehicle Interface situations, while providing a uniform application layer interface to the ATO Onboard System.

Chapter 7 Additional/ Optional Data exchanged with TCMS (in addition to Subset 139), describes the data which are not part of Subset 139, but that are defined for specific projects.

8 Appendix: Discussion of some specific usage scenarios provides some non-exhaustive and no-normantive examples of cases and combination of systems.

2.6 Relation to other documents

This document builds on the Train Interface FFFIS [27] and on the OCORA Train Interface Architecture Document [10].

The intention is to provide design guidance for integration of ATO and the vehicle interface.
It extends the scope of Subset-139 to non- standard and legacy vehicles and aims to provide a uniform interface for ATO as far as possible.

This document is intended to be used in conjunction with Subset-139 [17]. In case of doubt, Subset 139 [17] shall not be violated by this document. But feedback shall on this specification.

2.7 How to use this specification

2.7.1.1 This document provides an overview of the interface for accessing the functions and data provided through the TCMS. For this purpose, an overview of the architecture and the data (structured in packets, variables and hard- wired signals) is given.

2.7.1.2 The interface as described in this document provides standardised access to the Subset-139 FFFIS.

2.7.1.3 The definitive specification of the FVA is provided in the form of a model and the derived formal documentation. This document provides the basic information that is required as an entry point into the model.

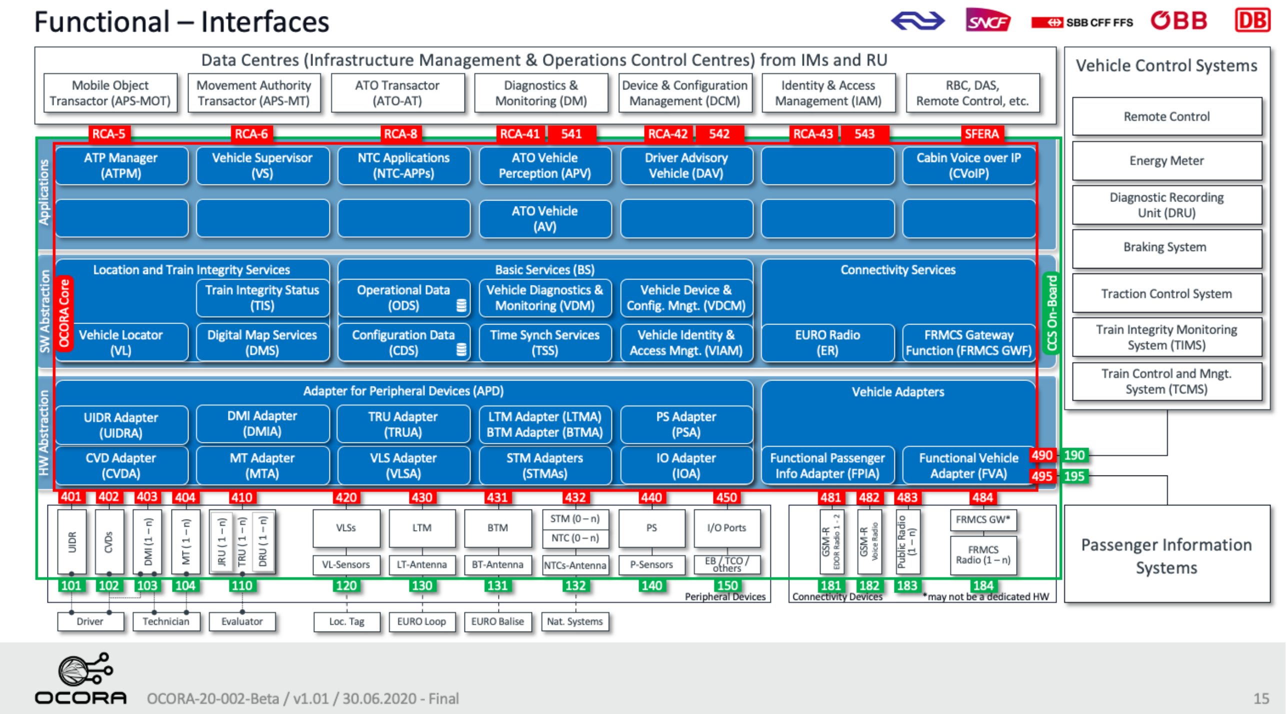
2.7.1.4 No changes to the ATO shall be required in order to connect it to a vehicle. For vehicles that have a fully Subset-139 [17] - compliant TCMS, the ATO-TCMS standardised interface will work without changes to the default parameter set. In case of gaps (the EVC does not support all data sent and received by the TCMS, and/ or the TCMS doesn’t support the standard set of packets covered by Subset-139 [17], it may be required to adapt the parameters of the FVA and/ or to add Specific Vehicle Interface functionality in the form of a Specific Vehicle Interface system or Specific Vehicle Interface software modules.

3 Architecture

3.1 Introduction

3.1.1 Context

The Remote API for ATO- TCMS connection is to be seen in context of the emerging OCORA Architecture.



Functional – Interfaces

SBB CFF FFS ÖBB DB SNCF

Data Centres (Infrastructure Management & Operations Control Centres) from IMs and RU

Mobile Object Transactor (APS-MOT)
Movement Authority Transactor (APS-MT)
ATO Transactor (ATO-AT)
Diagnostics & Monitoring (DM)
Device & Configuration Management (DCM)
Identity & Access Management (IAM)
RBC, DAS, Remote Control, etc.

RCA-5 RCA-6 RCA-8 RCA-41 541 RCA-42 542 RCA-43 543 SFERA

Applications

ATP Manager (ATPM)
Vehicle Supervisor (VS)
NTC Applications (NTC-APPs)
ATO Vehicle Perception (APV)
Driver Advisory Vehicle (DAV)
Cabin Voice over IP (CVoIP)
ATO Vehicle (AV)

SW Abstraction

OCORA Core

Location and Train Integrity Services
Train Integrity Status (TIS)
Vehicle Locator (VL)
Digital Map Services (DMS)
Basic Services (BS)
Operational Data (ODS)
Vehicle Diagnostics & Monitoring (VDM)
Vehicle Device & Config. Mngt. (VDCM)
Configuration Data (CDS)
Time Synch Services (TSS)
Vehicle Identity & Access Mngt. (VIAM)
Connectivity Services
EURO Radio (ER)
FRMCS Gateway Function (FRMCS GWF)

CCS On-Board

HW Abstraction

Adapter for Peripheral Devices (APD)
UIDR Adapter (UIDRA)
DMI Adapter (DMIA)
TRU Adapter (TRUA)
LTM Adapter (LTMA) BTM Adapter (BTMA)
PS Adapter (PSA)
CVD Adapter (CVDA)
MT Adapter (MTA)
VLS Adapter (VLSA)
STM Adapters (STMAs)
IO Adapter (IOA)
Vehicle Adapters
Functional Passenger Info Adapter (FPIA)
Functional Vehicle Adapter (FVA)

490 190 495 195

401 402 403 404 410 420 430 431 432 440 450 481 482 483 484

UIDR
CVDs
DMI (1 – n)
MT (1 – n)
JRU (1 – n)
TRU (1 – n)
DRU (1 – n)
VLSs
VL-Sensors
LTM
LT-Antenna
BTM
BT-Antenna
STM (0 – n)
NTC (0 – n)
NTCs-Antenna
PS
P-Sensors
I/O Ports
EB / TCO / others
Peripheral Devices

GSM-R EDOR Radio 1 - 2
GSM-R Voice Radio
Public Radio (1 – n)
FRMCS GW*
FRMCS Radio (1 – n)
Connectivity Devices
*may not be a dedicated HW

101 102 103 104 110 120 130 131 132 140 150 181 182 183 184

Driver
Technician
Evaluator
Loc. Tag
EURO Loop
EURO Balise
Nat. Systems

Vehicle Control Systems
Remote Control
Energy Meter
Diagnostic Recording Unit (DRU)
Braking System
Traction Control System
Train Integrity Monitoring System (TIMS)
Train Control and Mngt. System (TCMS)

Passenger Information Systems

OCORA OCORA-20-002-Beta / v1.01 / 30.06.2020 - Final 15

Figure 2 OCORA Overview OSI Layer 7

3.1.1.1 In OCORA architecture, the Automated Vehicle System (OCORA AV) will access the standardized interface through interfaces 490 (in order to access the Braking System and the Traction Control System, as well the Train Control and Management System.

3.1.1.2 In context of TSI 2022, the ATO- TCMS standardised interface is part of the CCS- TCMS interface. Note: The OCORA architecture and its specification are not finally defined yet.

3.1.1.3 The EVC API provides an interface presented to the ATO.

3.1.1.4 The FVA takes the inputs from the ATO API. Depending on the ParameterSet, the data may be forwarded to the Subset-139 FFFIS, forwarded to the Specific Vehicle Interface or discarded.

3.1.1.5 The FVA takes the inputs from the Specific Vehicle Interface and the FFFIS. Depending on the ParameterSet, the data will be forwarded to the ATO.

3.1.1.6 The FVA provides a stateful view of the data to all participants.

3.1.1.7 The OCORA conformal interface shall be structured in functional layers

3.1.1.8 The ATO core interface provides an application- layer interface to the ATO application as defined in UNISIG Subset-125.

3.1.1.9 The Functional Vehicle Interface provides a standardized interface, while allowing configuration for project- specific data without change to the connected devices.

3.1.1.10 Note: Some functions may be implemented using a separate application or system, called “Specific Vehicle Interface” here.

3.1.1.11 The interface to TCMS (= interface to Subset-139 common definition) is responsible for assuring plug and play functionality with any underlying TCMS.

3.1.1.12 Note: The lower layers of Subset-139 need to be adapted for each project.

3.1.1.13 Note: Use of the Universal Vital Command and Control Bus will ensure transparent Plug & Play functionality for all projects at a later stage.

4 ATO Core Interface

4.1 General

4.1.1 Odometry information

4.1.1.1 ATO-OB may implement its own odometry to calculate the train position and speed as required by [16].

4.1.1.2 It shall be possible to use information provided by already existing sensors, components or systems as an input value for ATO-OB odometry. This will enable the system integrator at the train level to make an optimal configuration under the consideration of specific vehicle type and its characteristics.

4.1.1.3 ATO-OB can optionally use the odometry information provided by TCMS (as “already existing system” according to 5.2.3.1), if this information is of sufficient quality. The required quality of information is project specific.

4.2 Quality of Service

4.2.1 Principles

The properties of packets are expressed as quality of service (QoS).

The following qualities can be expressed

4.2.1.1 Bandwidth

Bandwidth refers to the data rate that can be transmitted within 1s and is expressed in kBytes.

4.2.1.2 Delay

Maximum delay between availability of a set of data at the sender and its reception at the receiver.

4.2.1.2.1 Note: This is also applicable to publish/ subscribe architectures

4.2.1.3 Integrity

The reliability of data transport.

4.2.1.4 Safety

Safety Requirements

4.2.1.5 Persistence

The lifetime of the data.

4.2.1.5.1 Note: This approach is in preparation for future CCS systems

4.3 ATO Vehicle Adapter: Packets from AV to TCMS

4.3.1 Packet Number 0: ATO Status

Packet ID	0		
Description	ATO Status message		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	ATO_STATE_ACPU	3	ATO state, SS139 6.2.2.1 Table 2
	ATO_CONFIG_ACPU	2	ATO status, SS139 6.2.2.1 Table 2

4.3.2 Packet Number 1: Propulsion (Traction / Dynamic Brake) Control

Packet ID	1		
Description	Propulsion (Traction / Dynamic Brake) Control commands		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	RELATIVE_TRACTION_REQUEST_ACPU	11	Relative traction / brake request, SS139 6.2.3.1 Table 3
	TRACTION_REQUEST_ACPU	1	Traction request, SS139 6.2.3.1 Table 3
	BRAKE_REQUEST_ACPU	1	Brake request, SS139 6.2.3.1 Table 3

4.3.3 Packet Number 2: Pneumatic and special brake control

Packet ID	2		
Description	Pneumatic and special brake control commands		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	INDIRECT_BRAKE_REQUEST_ACPU	11	Immediate indirect air brake request, SS139, 6.2.4.8, Table 4a
	DIRECT_BRAKE_REQUEST_ACPU	11	Immediate direct air brake request, SS139, 6.2.4.8, Table 4a
	RELEASE_QUICK_BRAKE_ACPU	1	Quick brake release request, SS139, 6.2.4.8, Table 4a

4.3.4 Packet Number 3: Holding Brake control

Packet ID	3		
Description	Holding Brake control command		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	HOLDING_BRAKE_REQUEST_ACPU	1	Holding brake request, SS139, 6.2.4.12, Table 4b

4.3.5 Packet Number 5: Door control

Packet ID	5		
Description	Door control commands		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	DOOR_ENABLE_REQUEST_ACPU	4	Door enable request, SS139, 6.2.6.1, Table 6
	DOOR_OPEN_REQUEST_ACPU	2	Door open request, SS139, 6.2.6.1, Table 6
	DOOR_CLOSE_REQUEST_ACPU	2	Door close request, SS139, 6.2.6.1, Table 6

4.3.6 Packet Number 9: Config Info Request

Packet ID	9		
Description	Request TCMS capabilities packet		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	TCMS_CAPABILITIES_REQUEST_ACPU	1	

4.3.7 Packet Number 10: ATO Time

Packet ID	10		
Description	ATO UTC Time information.		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	UTC_TIME_ACPU	32	
	UTC_TIME_MS_ACPU	32	

4.4 ATO Vehicle Adapter: Packets from FVA to AV

4.4.1 Packet Number 21: Propulsion (Traction / Dynamic Brake) Status

Packet ID	21		
Description	Propulsion (Traction / Dynamic Brake) Status		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	UTC_TIME_TCMS	32	Timestamp
	UTC_TIME_MS_TCMS	32	Timestamp
	TRACTION_READY_TCMS	1	Traction ready, SS139 6.2.3.1 Table 3
	ENGAGEMENT_READY_TCMS	1	Engagement ready, SS139 6.2.3.1 Table 3
	TRACTION_APPLIED_TCMS	1	Traction applied, SS139 6.2.3.1 Table 3
	DYNAMIC_BRAKE_READY_TCMS	1	Dynamic brake ready, SS139 6.2.3.1 Table 3
	DYNAMIC_BRAKE_APPLIED_TCMS	1	Dynamic brake applied, SS139 6.2.3.1 Table 3
	EB_RELEASED_TCMS	1	EB released, SS139, 6.2.4.1, Table 4

Packet ID	21		
	SB_APPLIED_TCMS	1	SB applied, SS139, 6.2.4.1, Table 4
	TRACTION_OVER_BRAKE_ENABLED_TCMS	1	Traction over brake enabled, SS139, 6.2.4.8, Table 4

4.4.2 Packet Number 22: Pneumatic and special brake Status

Packet ID	22		
Description	Pneumatic and special brake Status		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	BRAKE_PIPE_PRESSURE_TCMS	10	Brake pipe pressure
	BRAKE_DISTRIBITOR_PRESSURE_TCMS	10	Pressure at brake distributor output
	DIRECT_BRAKE_APPLIED_TCMS	1	Direct brake applied
	EQUALISING_RES_PRESSURE_TCMS	10	Optional

4.4.3 Packet Number 23: Holding Brake status

Packet ID	23		
Description	Holding Brake status		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	HOLDING_BRAKE_APPLIED_TCMS	1	Holding brake status, SS139, 6.2.4.12, Table 4b

4.4.4 Packet Number 24: Brake Model

Packet ID	24		
Description	Model of the emergency brake, traction, and service brake (if preQoS), to be used by the Core CPU		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	Q_BRAKE_MODEL_TCMS	1	The following fields only if Q_BRAKE_MODEL = 1
	MODEL_BEGIN_BRAKE_TCMS	8	Part of EB model
	MODEL_FULL_BRAKE_TCMS	11	Part of EB model
	N_ITER	5	Part of EB model max. value: 5
	MODEL_SPEED_TCMS(k)	8	Part of EB model
	MODEL_DECELER_TCMS(k)	8	Part of EB model
	CUT_TRACT_DELAY_TCMS	8	Part of traction model
	TRAIN_MAX_ACC_TCMS	10	Part of traction model
	ACC_COEF_SB_UNUSED_TCMS	2	Part of traction model
	ACC_COEF_SB_USED_TCMS	2	Part of traction model
	Q_SB_MODEL_PREQOS	1	Part of SB model
	MODEL_BEGIN_BRAKE_TCMS	8	Part of SB model
	MODEL_FULL_BRAKE_TCMS	11	Part of SB model
	N_ITER	5	Part of SB model max. value: 5
	MODEL_SPEED_TCMS(k)	8	Part of SB model
	MODEL_DECELER_TCMS(k)	8	Part of SB model
	MIN_ROT_MASS_PERCENT_TCMS	8	Part of rot mass model
	NOM_ROT_MASS_PERCENT_TCMS	8	Part of rot mass model
	MAX_ROT_MASS_PERCENT_TCMS	8	Part of rot mass model
	T_W_TCMS	13	Part of driver delay
	T_P_TCMS	13	Part of driver delay

Packet ID	**24**		
	T_I_P_TCMS	13	Part of driver delay
	T_RSMA_TCMS	13	Part of driver delay

4.4.5 Packet Number 25: Odometry Data

Packet ID	**25**		
Description	Odometry data		
QoS	1		
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	UTC_TIME_TCMS	32	TCMS timestamp
	UTC_TIME_MS_TCMS	32	TCMS timestamp
	ACTUAL_SPEED_TCMS	18	Actual speed: SS139, §6.2.5.2 Table 5;
	ACTUAL_ACCELERATION_TCMS	15	Actual acceleration: SS139, §6.2.5.2 Table 5
	TRAVELLED_DISTANCE_TCMS	32	Travelled distance: SS139, §6.2.5.2 Table 5
	TSI_STANDSTILL_TCMS	1	TSI standstill: SS139, §6.2.5.2 Table 5
	DOOR_OPENING_PERMITTED_TCMS	1	Optional variable

4.4.5.1 The related packet see SS139, § 7.3.4, Table 11

4.4.6 Packet Number 26: Door status

Packet ID	**26**		
Description	Door status data		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	DOOR_STATUS_TCMS	16	Door status signals, SS139, 6.2.6.1, Table 6

4.4.7 Packet Number 27: Train and vehicle specific values

Packet ID	**27**		
Description	Train and vehicle specific values		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	Q_MAX_AVAILABLE_TRACTIVE_EFFORT_TCMS	1	
	MAX_AVAILABLE_TRACTIVE_EFFORT_TCMS	12	Maximum available tractive effort (for the whole train)
	Q_MAX_AVAILABLE_TRACTIVE_POWER_TCMS	1	
	MAX_AVAILABLE_TRACTIVE_POWER_TCMS	15	Maximum available tractive output power (for the whole train)
	Q_AVAILABLE_TRACTIVE_EFFORT_TCMS	1	
	AVAILABLE_TRACTIVE_EFFORT_TCMS	12	Currently available tractive effort (for the whole train)
	Q_MAX_AVAILABLE_DYNAMICBRAKE_EFFORT_TCMS	1	
	MAX_AVAILABLE_DYNAMICBRAKE_EFFORT_TCMS	12	Maximum available dynamic brake effort (for the whole train)
	Q_AVAILABLE_DYNAMICBRAKE_EFFORT_TCMS	1	

Packet ID	**27**		
	AVAILABLE_DYNAMICBRAKE_EFFORT_TCMS	12	Currently available dynamic brake effort (for the whole train)
	Q_MAX_AVAILABLE_DYNAMICBRAKE_POWER_TCMS	1	
	MAX_AVAILABLE_DYNAMICBRAKE_POWER_TCMS	15	Maximum available dynamic brake power (for the whole train)
	Q_TRAIN_MASS_TCMS	1	
	TRAIN_MASS_TCMS	14	Train mass
	MAX_TRAIN_SPEED_TCMS	8	Max Train Speed
	BRAKE_MODE_TCMS	2	Brake mode
	N_ITER	5	
	WHEEL_DIAMETER_TCMS	16	Wheel diameters

Packet Number 28: Train and vehicle specific values (fast)Packet ID	**28**		
Description	Train and vehicle specific values		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	ACTUAL_INPUTCURRENT_TCMS	15	Actual input current
	TB_SET_TCMS	13	T/B set value
	ADHESIONFACTOR_REDUCTION_TCMS	7	Adhesion factor reduction
	TB_LEVER_TCMS	2	T/B lever position
	TB_LEVER_FAILURE_TCMS	1	
	BRAKE_POSITION_TCMS	1	
	N_ITER	1	N_ITER for Speed_Sensor_Error_TCMS
	SPEED_SENSOR_STATUS_TCMS	4	
	SPEED_SENSOR_PULSES_TCMS	20	Pulses per km

4.4.8 Packet Number 29: UTC Master Time

Packet ID	**29**		
Description	UTC Time information.		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	UTC_TIME_TCMS	32	
	UTC_TIME_MS_TCMS	32	
	UTC_MASTER_TCMS	2	
	TIME_OFFSET_SIGN_TCMS	3	
	TIME_OFFSET_TCMS	32	

4.4.9 Packet Number 31: TCMS Capabilities

Packet ID	**28**		
Description	TCMS Capabilities		
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	CPB_AFB_Speed_Installed	1	
	CPB_AFB_Traction_Installed	1	
	CPB_Brake_Blending_Installed	1	
	CPB_Brake_Model_cfg	1	
	CPB_Dynamic_Brake_Installed	1	
	CPB_Engagement_Ready_cfg	1	
	CPB_HoldingBrakeApplied_cfg	1	

Packet ID	28		
	CPB_Traction_Ready_cfg	1	
	CPB_TractionApplied_cfg	1	
	CPB_Full_Ocora	1	
	CPB_Standard_139	1	

4.4.9.1 This packet is generated by the FVA. It contains data concerning FVA configuration.

4.4.10 Packet Number 32: Error Status

Packet ID	32		
Description			
QoS			
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	ERROR_BRAKEREQUEST_NOT_CFG	1	
	ERROR_DOORCONTROL_NOT_CFG	1	
	ERROR_DOORENABLE_NOT_CFG	1	
	ERROR_HOLDINGBRAKE_NOT_CFG	1	
	ERROR_RELINDIRECTBRAKE_NOT_CFG	1	
	ERROR_TRACTION_OPTION_1_NOT_CFG	1	
	ERROR_TRACTION_OPTION_2_NOT_CFG	1	
	ERROR_TRACTIONREQUEST_NOT_CFG	1	
	ERR_RELQUICKBRAKE_NOT_CFG	1	

4.4.10.1 This packet is generated by the FVA. It contains data concerning FVA error status.

4.5 Variables

4.5.1.1 The variable names are derived from the names as defined in Subset-139 as far as appropriate.

4.5.1.2 We use the following prefixes and suffixes to help to identify the scope of the variables:

- _ACPU: Sent by the ATO core processing unit
- CPB_: Descriptor for TCMS capability
- ERR_: Error
- _TCMS Sent by the train interface unit
- _DMOD_: part of the dynamic models

4.5.1.3 The variables are listed alpabetically. However, the sorting ignores the prefixes, so that each variable can easily be found by its name as known from Subset-139.

4.5.2 ACTUAL_ACCELERATION_TCMS

Name	**ACTUAL_ACCELERATION_TCMS**		
Description	Actual acceleration Value from TCMS Range: -3500 … 0 … +3500 mm/s^2, resolution: 1 mm/s^2 See [24] 4.2.4.5.1 (5)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 15 bit	-3500 mm/s^2 11110010 01010100	3500 mm/s^2 00001101 10101100	1 mm/s^2 BCD

4.5.2.1 ATO-OB uses acceleration information for on-track localisation, for computing speed profiles and for train control.

4.5.3 ACTUAL_INPUTCURRENT_TCMS

Name	**ACTUAL_INPUTCURRENT_TCMS**		
Description	Actual input current Actual value of input current (for the whole train) Range: - 10 000 A… 0 … + 10 000 A, resolution <= 1 A (10 A @ DC systems) (negative values refer to regenerative brake current) Note: ATO-OB uses this variable for maintaining the track condition “limitation of input current”. Note: Negative values (regenerative braking) are mandatory for Locos, for EMUs they are optional.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 15 bit	-10000 -10000	10000 10000	1

4.5.4 ACTUAL_SPEED_TCMS

Name	**ACTUAL_SPEED_TCMS**		
Description	Actual speed Value from TCMS Range: 0 … 166 667 mm/s (600 km/h), resolution 1 mm/s (ATO format)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 18 bit	0 mm/s 0	166 667 mm/s 166 667	1 mm/s

4.5.4.1 ATO-OB uses speed information for on-track localisation, for computing speed profiles and for train control.

4.5.5 ADHESIONFACTOR_REDUCTION_TCMS

Name	**ADHESIONFACTOR_REDUCTION_TCMS**		
Description	Adhesion factor reduction Reduction of adhesion (for informing ATO-TS) Values 0 and 1 for reporting the bad adhesion according to ETCS convention. Values: 10 (really bad adhesion) … 100 % (full adhesion, no limitation), are reserved for future use.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 7 bit	0	100	1

4.5.6 ATO_CONFIG_ACPU

Name	**ATO_CONFIG_ACPU**		
Description	ATO Configuration Information Identifies the ATO configuration - output signal type (Option I or Option II)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 2 bit			
Special/ Reserved Values	0	No Option selected	
	1	output signal type Option I	
	2	output signal type Option II	

4.5.6.1 Note: The ATO config signal identifies how the TCMS will interpret the ATO-OB output signal Relative traction / brake request - whether the Option I or Option II is used (see later in this Subset).

4.5.7 ATO_STATE_ACPU

Name	**ATO_STATE_ACPU**		
Description	ATO State Information Values NP, CO, NA, AV, RE, EG, DE, FA correspond to particular states of ATO-OB.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 3 bit			
Special/ Reserved Values	0	ATO_STATE_NP	NP
	1	ATO_STATE_CO	CO
	2	ATO_STATE_NA	NA
	3	ATO_STATE_AV	AV
	4	ATO_STATE_RE	RE
	5	ATO_STATE_EG	EG
	6	ATO_STATE_DE	DE
	7	ATO_STATE_FA	FA

4.5.7.1 ATO state is sporadic information which is only sent when it changes and upon initialisation.

4.5.7.2 Note: The TCMS uses ATO state signal to decide which ATO-OB output signals from the list of ATO active functions shall be followed and which ATO-OB input signals shall be generated (see [16] Chapt. 9.11).

4.5.8 AVAILABLE_DYNAMICBRAKE_EFFORT_TCMS

Name	**AVAILABLE_DYNAMICBRAKE_EFFORT_TCMS**		
Description	Currently available dynamic brake effort (for the whole train) Max. dynamic brake effort at current speed. Includes both multiple traction and reduced dynamic brake capabilities (isolated bogie etc.) Range: 0 ... 3000 kN, resolution 1 kN, Only if Q_ Available_DynamicBrake_Effort = 1		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 12 bit	0 kN 0	3000 kN 3000	1 kN

4.5.9 AVAILABLE_TRACTIVE_EFFORT_TCMS

Name	**AVAILABLE_TRACTIVE_EFFORT_TCMS**		
Description	Currently available tractive effort (for the whole train) Maximum tractive effort at current speed. Includes both multiple traction and reduced traction capabilities (isolated bogie etc.) Range: 0 ... 3000 kN, resolution 1 kN Only if Q_ Available_Traction_Effort = 1		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 12 bit	0 kN 0	3000 kN 3000	1 kN

4.5.10 BRAKE_DELAY_CLASS_ID_ACPU

Name	**BRAKE_DELAY_CLASS_ID_ACPU**		
Description	Brake delay class ID		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer	0	255	1

Name	BRAKE_DELAY_CLASS_ID_ACPU		
8 bits used	0	255	

4.5.11 BRAKE_DISTRIBITOR_PRESSURE_TCMS

Name	**BRAKE_DISTRIBITOR_PRESSURE_TCMS**		
Description	Pressure at brake distributor output 0 … 10 bar, resolution ≤ 0.05 bar. Necessary when ATO controls the brake force splitting and/or brake blending.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 10 bit	0.00 bar 0	10.00 bar 1000	0.01 bar

4.5.11.1 This variable is mandatory for Locos and optional (project-specific) for EMUs.

4.5.12 BRAKE_MODE_TCMS

Name	**BRAKE_MODE_TCMS**		
Description	Brake mode Mandatory for Locos: G / P / R / +Ep Note: R+Mg is not relevant for ATO.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 2 bit			
Special/ Reserved Values	0	G	
	1	P	
	2	R	
	3	+Ep	

4.5.13 BRAKE_PIPE_PRESSURE_TCMS

Name	**BRAKE_PIPE_PRESSURE_TCMS**		
Description	Brake pipe pressure 0 … 10 bar, resolution ≤ 0.05 bar. Necessary when ATO controls the brake force splitting and/or brake blending.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 10 bits	0.00 bar 0	10.00 bar 1000	0.01 bar

4.5.13.1 This variable is mandatory for Locos and optional (project-specific) for EMUs.

4.5.13.2 Note: As this signal enables the instant control of air brake, they also allow the forced use of air brake according to national rules.

4.5.13.3 Note: If Direct brake is requested (by *Immediate direct air brake request*) and not confirmed by *Direct brake applied* signal, then ATO-OB will request service (indirect) brake instead (to be included in SS-125).

4.5.14 BRAKE_POSITION_TCMS

Name	**BRAKE_POSITION_TCMS**		
Description	Brake Lever Position information		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned integer 2 bits			
Special/ Reserved Values	0	all brake levers in zero positions	

	1	any of brake levers is out of neutral position	
	2	Unknown	

4.5.15 BRAKE_REQUEST_ACPU

Name	**BRAKE_REQUEST_ACPU**		
Description	Auxiliary control signal for dynamic brake control		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	No Brake requested	
	1	Brake requested	

4.5.15.1 Brake request is a mandatory signal. It is processed by the functional vehicle interface.

4.5.15.2 Brake request is a cyclic signal.

4.5.15.3 Brake request corresponds to Driveline engaged signal according to [24] in traction (Traction applied) or brake (Dynamic brake applied) modes.

4.5.15.4 There exist two options of interpretation of Relative traction/brake request signal. The decision of which option will be used is ATO-OB-supplier's specific. The TCMS may provide either option. If no option is provided, then the ATO must control the traction/ brake directly using low-level commands.

4.5.16 CPB_AFB_SPEED_INSTALLED

Name	**CPB_AFB_SPEED_INSTALLED**		
Description	AFB (speed setting) installed TCMS is Automatischer Fahrbetrieb capable (speed preset)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	AFB (speed setting) not installed	
	1	AFB (speed setting) installed	

4.5.17 CPB_AFB_TRACTION_INSTALLED

Name	**CPB_AFB_TRACTION_INSTALLED**		
Description	Capabilty information: AFB (traction setting) installed TCMS is Automatischer Fahrbetrieb capable (traction preset)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	AFB (traction setting) not installed	
	1	AFB (traction setting) installed	

4.5.18 CPB_BRAKE_BLENDING_INSTALLED

Name	**CPB_BRAKE_BLENDING_INSTALLED**
Description	Capability information Brake Blending installed The TCMS is capable of doing brake blending

Name	**CPB_BRAKE_BLENDING_INSTALLED**		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Brake Blending not installed	
	1	Brake Blending installed	

4.5.19 CPB_BRAKE_MODEL_CFG

Name	**CPB_BRAKE_MODEL_CFG**		
Description	Capability information Brake model present A brake model is available (from TCMS or from the Functional Vehicle Adaptor)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Brake model not present	
	1	Brake model present	

4.5.20 CPB_DYNAMIC_BRAKE_INSTALLED

Name	**CPB_DYNAMIC_BRAKE_INSTALLED**		
Description	Capability information Dynamic Brake installed		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Dynamic Brake not installed	
	1	Dynamic Brake installed	

4.5.21 CPB_ENGAGEMENT_READY_NOT_CFG

Name	**CPB_ENGAGEMENT_READY_NOT_CFG**		
Description	Capability information Engagement Ready not present Engagement not present in TCMS. Sent during startup		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Engagement Ready not present	
	1	Engagement Ready not present	

4.5.22 CPB_Full_Ocora

Name	**CPB_FULL_OCORA**		
Description	Capability information The TCMS is fully OCORA compliant		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	The TCMS is not OCORA compliant	
	1	The TCMS is fully OCORA compliant	

4.5.23 CPB_HOLDINGBRAKEAPPLIED_NOT_CFG

Name	**CPB_HOLDINGBRAKEAPPLIED_NOT_CFG**		
Description	Capability information Holding brake applied signal not present		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Holding brake applied signal present	
	1	Holding brake applied signal not present	

4.5.24 CPB_Standard_139

Name	**CPB_STANDARD_139**		
Description	Capability information The TCMS is fully Subset-139 compliant		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	The TCMS is not Subset-139 compliant	
	1	The TCMS is fully Subset-139 compliant	

4.5.25 CPB_TRACTION_READY_NOT_CFG

Name	**CPB_TRACTION_READY_NOT_CFG**		
Description	Capability information Traction Ready not present Traction Ready not present in TCMS. Sent during startup		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Traction Ready present	
	1	Traction Ready not present	

4.5.26 CPB_TRACTIONAPPLIED_NOT_CFG

Name	**CPB_TRACTIONAPPLIED_NOT_CFG**		
Description	Capability information Traction applied not present Traction applied not present in TCMS. Sent during startup		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Traction applied present	
	1	Traction applied not present	

4.5.27 DECELERATION_CLASS_ID_ACPU

Name	**DECELERATION_CLASS_ID_ACPU**		
Description	Deceleration class ID		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Integer 8 bits	0	255	1

4.5.28 DIRECT_BRAKE_APPLIED_TCMS

Name	**DIRECT_BRAKE_APPLIED_TCMS**		
Description	Traction over brake enabled Feedback signal - the vehicle braked by Direct brake.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Direct brake not applied	
	1	Direct brake applied	

4.5.28.1 This variable is mandatory for Locos and optional (project-specific) for EMUs.

4.5.28.2 Note: As this signal enables the instant control of air brake, they also allow the forced use of the air brake according to national rules.

4.5.28.3 Note: If Direct brake is requested (by *Immediate direct air brake request*) and not confirmed by the *Direct brake applied* signal, then ATO-OB will request the service (indirect) brake instead (to be included in SS-125).

4.5.29 DIRECT_BRAKE_REQUEST_ACPU

Name	**DIRECT_BRAKE_REQUEST_ACPU**		
Description	Immediate direct air brake request Auxiliary control signal for direct control of direct (Locomotive) air brake Range: 0 … 100% (full direct brake), resolution ≤ 1%		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 11 bit	0 % 0	100.0% 1000	0,1%

4.5.29.1 Note: As this signal enables the instant control of air brake, they also allow the forced use of air brake according to national rules.

4.5.29.2 Note: If Direct brake is requested (by *Immediate direct air brake request*) and not confirmed by the *Direct brake applied* signal, then ATO-OB will request service (indirect) brake instead (to be included in SS-125).

4.5.30 DMOD_ACC_COEF_SB_UNUSED_TCMS

Name	**DMOD_ACC_COEF_SB_UNUSED_TCMS**		
Description	Acceleration coefficient when the service brake is not present or not available. Ponderation coefficient to be applied on maximum train acceleration when the service brake is not available.		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 7 bits	0 0	1 100	0,01

4.5.31 DMOD_ACC_COEF_SB_USED_TCMS

Name	**DMOD_ACC_COEF_SB_USED_TCMS**		
Description	Acceleration coefficient when the service brake is available. Ponderation coefficient to be applied on maximum train acceleration acceleration when the service brake is available.		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 7 bits	0 0	1 100	0,01

4.5.32 DMOD_CUT_TRACT_DELAY_TCMS

Name	**DMOD_CUT_TRACT_DELAY_TCMS**		
Description	Delay to cut off traction Delay between the ordering of traction cut off and the effective cut off of the traction		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bits	0 s 0	25,5 s 255	0,1 s

4.5.33 DMOD_MAX_ROT_MASS_PERCENT_TCMS

Name	**DMOD_MAX_ROT_MASS_PERCENT_TCMS**		
Description	maximum rotating mass percentage maximum rotating mass of the train, expressed as a percentage of the total weight of the train		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bit	0% 0	25,5 % 255	0,1 %

4.5.34 DMOD_MIN_ROT_MASS_PERCENT_TCMS

Name	**DMOD_MIN_ROT_MASS_PERCENT_TCMS**		
Description	minimum rotating mass percentage minimum rotating mass of the train, expressed as a percentage of the total weight of the train		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bit	0 % 0	25,5 % 255	0,1 %

4.5.35 DMOD_MODEL_BEGIN_BRAKE_TCMS

Name	**DMOD_MODEL_BEGIN_BRAKE_TCMS**		
Description	Delay between ordering a brake application, and when brake begins to be applied (more than 0%)		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bit	0 s 0	25,5 s 255	0,1 s

4.5.36 DMOD_MODEL_DECELER_TCMS

Name	**DMOD_MODEL_DECELER_TCMS**		
Description	Brake model deceleration point Coordinate on the Y axis (=train deceleration) of a point of the deceleration model		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bit	0 m/s2 0	25,5 m/s2 255	0,1 m/s2

4.5.37 DMOD_MODEL_FULL_BRAKE_TCMS

Name	**DMOD_MODEL_FULL_BRAKE_TCMS**		
Description	Delay between when the braking effort begins (>0%) and when the full braking effort is reached (100%)		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 11 bits	0 s 0	120, 0 s 1200	0,1 s

4.5.38 DMOD_MODEL_SPEED_TCMS

Name	**DMOD_MODEL_SPEED_TCMS**		
Description	Brake model speed point Coordinate on the X axis (=train speed) of a point of the deceleration model		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Integer 8 bit	0 km/h 0	600 km/h 120	5 km/h

4.5.39 DMOD_NOM_ROT_MASS_PERCENT_TCMS

Name	**DMOD_NOM_ROT_MASS_PERCENT_TCMS**		
Description	nominal rotating mass of the train, expressed as a percentage of the total weight of the train		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Integer 8 bit	0% 0	25,5 % 255	0,1 %

4.5.40 DMOD_T_I_P_TCMS

Name	**DMOD_T_I_P_TCMS**		
Description	T_i_p parameter used by the ATO in the braking curve calculation		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 13 bits	0 s 0	600 s 6000	0,1s

4.5.41 DMOD_T_P_TCMS

Name	**DMOD_T_P_TCMS**		
Description	T_p parameter used by the ATO in the braking curve calculation		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 13 bits	0 s 0	600 s 6000	0,1s

4.5.42 DMOD_T_RSMA_TCMS

Name	**DMOD_T_RSMA_TCMS**		
Description	T_rsma parameter used by the ATO in the braking curve calculation		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 13 bits	0 s 0	600 s 6000	0,1s

4.5.43 DMOD_T_W_TCMS

Name	**DMOD_T_W_TCMS**		
Description	T_w parameter used by the ATO in the braking curve calculation		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 13 bits	0 s 0	600 s 6000	0,1s

4.5.44 DMOD_TRAIN_MAX_ACC_TCMS

Name	**DMOD_TRAIN_MAX_ACC_TCMS**
Description	Maximum acceleration that the train is able to reach

Name	DMOD_TRAIN_MAX_ACC_TCMS		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 10 bits	0 m/s^2 0	10,23 m/s^2 1023	0,01 m/s^2

4.5.45 DOOR_CLOSE_REQUEST_ACPU

Name	**DOOR_CLOSE_REQUEST_ACPU**		
Description	Door close request Requests to close the doors centrally		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Bitset 2 bit			
Special/ Reserved Values	00	Do not close the doors	
	01	Close the doors on the left side	
	10	Close the doors on the right side	

4.5.46 DOOR_ENABLE_REQUEST_ACPU

Name	**DOOR_ENABLE_REQUEST_ACPU**		
Description	Door enable request These signals enable the passengers to open individual doors (side selective; inside/outside selective; Door Selective)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Bitset 4 bit			
Special/ Reserved Values	0000	Passenger Door Request Disabled	
	0001	Left side Passenger Door Request enabled	
	0010	Right side Passenger Door Request enabled	
	0100	Inside Passenger Door Request enabled	
	1000	Outside Passenger Door Request enabled	

4.5.46.1 Note: To enable a certain mode for passenger door request, the bits shall be combined.

4.5.46.2 This command is overridden by the ETCS door command as defined in [27].

4.5.46.3 This command is not considered as safety relevant.

4.5.47 DOOR_OPEN_REQUEST_ACPU

Name	**DOOR_OPEN_REQUEST_ACPU**		
Description	Door open request Requests to open the doors centrally; side selective.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Bitset 2 bit			
Special/ Reserved Values	00	Do not open the doors	
	01	Open the doors on the left side	
	10	Open the doors on the right side	

4.5.47.1 This command is overridden by the ETCS door command as defined in [27].

4.5.47.2 This command is not considered as safety relevant.

4.5.48 DOOR_OPENING_PERMITTED_TCMS

Name	**DOOR_OPENING_PERMITTED_TCMS**		
Description	Door opening permitted Logical information about standstill according to national rules (signal for permitting the door opening)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Door opening not permitted	
	1	Door opening permitted	

4.5.48.1 *Door opening permitted* information is used for functions related to standstill (for example holding brake control, door control etc.)

4.5.49 DOOR_STATUS_TCMS

Name	**DOOR_STATUS_TCMS**		
Description	Door status signals Feedback signal - the actual status of doors: closed&locked / unreleased / released / open		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 16 bit			
Special/ Reserved Values	0	Doors open	
	1	Doors closed and locked	
	2	Doors unreleased	
	3	Doors released	
	4	Door sensor error	
	5- 65535	Spare	

4.5.49.1 Note: Format on Subset-139 side not finally decided

4.5.50 DYNAMIC_BRAKE_APPLIED_TCMS

Name	**DYNAMIC_BRAKE_APPLIED_TCMS**		
Description	Dynamic brake applied Propulsion reports that dynamic brake is applied. For Locos and EMUs only.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Dynamic brake not applied	
	1	Dynamic brake applied	

4.5.51 DYNAMIC_BRAKE_AVAILABLE_TCMS

Name	**DYNAMIC_BRAKE_AVAILABLE_TCMS**		
Description	Dynamic brake available Dynamic brake is generally available		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Dynamic brake not available	
	1	Dynamic brake available	

4.5.51.1 This info needs to be provided by the ATO. It was agreed to add this signal to Subset-125

4.5.52 DYNAMIC_BRAKE_READY_TCMS

Name	**DYNAMIC_BRAKE_READY_TCMS**		
Description	Dynamic brake ready All conditions for applying the dynamic brake are fulfilled. If this signal is active, then ATO-OB is allowed to request the dynamic brake. For Locos and EMU only. Note: This signal stays false if no dynamic brake is installed		
Type	**Minimum Value**	**Maximum Value**	**Resolution/ Formula**
Boolean 1 bit			
Special/ Reserved Values	0	Dynamic brake not ready	
	1	Dynamic brake ready	

4.5.53 EB_RELEASED_TCMS

Name	**EB_RELEASED_TCMS**		
Description	Emergency Brake released Emergency brake not applied (brake pipe pressure >= 3.5 bar)		
Type	**Minimum Value**	**Maximum Value**	**Resolution/ Formula**
Boolean 1 bit			
Special/ Reserved Values	0	EB not released	
	1	EB released	

4.5.53.1 The *EB released* signal is mandatory for both Locos and EMUs.

4.5.54 ENGAGEMENT_READY_TCMS

Name	**ENGAGEMENT_READY_TCMS**		
Description	Engagement ready Explanation: All conditions for engagement are fulfilled (including door closed, direction selected, etc.). If this signal disappears, ATO disengages. When the signal re-appears, driver must push engage button for continuing in automated mode.		
Type	**Minimum Value**	**Maximum Value**	**Resolution/ Formula**
Boolean 1 bit			
Special/ Reserved Values	0	Engagement not ready	
	1	Engagement ready	

4.5.55 ERROR_BRAKEREQUEST_NOT_CFG

Name	**ERROR_BRAKEREQUEST_NOT_CFG**		
Description	Brake Request not present Brake Request request from ATO while not present in TCMS		
Type	**Minimum Value**	**Maximum Value**	**Resolution/ Formula**
Boolean 1 bit			
Special/ Reserved Values	0	Brake Request present	
	1	Brake Request not present	

4.5.56 ERROR_DOORCONTROL_NOT_CFG

Name	**ERROR_DOORCONTROL_NOT_CFG**
Description	No door control present

	Door command received from ATO whil no doors can be controlled on the train		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Door control present	
	1	No door control present	

4.5.57 ERROR_DOORENABLE_NOT_CFG

Name	**ERROR_DOORENABLE_NOT_CFG**		
Description	Door enable request not present Door enable request from ATO while not present in TCMS		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Door enable request present	
	1	Door enable request not present	

4.5.58 ERROR_HOLDINGBRAKE_NOT_CFG

Name	**ERROR_HOLDINGBRAKE_NOT_CFG**		
Description	Holding brake request not present Holding brake request from ATO while not present in TCMS		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Holding brake request present	
	1	Holding brake request not present	

4.5.59 ERROR_PNEUBRAKE_NOT_CFG

Name	**ERROR_PNEUBRAKE_NOT_CFG**		
Description	Pneumatic Brake Control not present Pneumatic Brake Control request from ATO while not present in TCMS		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Pneumatic Brake Control present	
	1	Pneumatic Brake Control not present	

4.5.60 ERROR_RELINDIRECTBRAKE_NOT_CFG

Name	**ERROR_RELINDIRECTBRAKE_NOT_CFG**		
Description	Relative immediate Indirect Brake Request not present Relative Immediate Indirect Brake request from ATO while not present in TCMS		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			

Name	ERROR_RELINDIRECTBRAKE_NOT_CFG		
Special/ Reserved Values	0	Relative immediate Indirect Brake Request present	
	1	Relative immediate Indirect Brake Request not present	

4.5.61 ERROR_RELQUICKBRAKE_NOT_CFG

Name	ERROR_RELQUICKBRAKE_NOT_CFG		
Description	Quick brake release request not present Quick brake release request from ATO while not present in TCMS		
Type	*Minimum Value*	*Maximum Value*	*Resolution/ Formula*
Boolean 1 bit			
Special/ Reserved Values	0	Quick brake release request present	
	1	Quick brake release request not present	

4.5.62 ERROR_TRACTION_OPTION_1_NOT_CFG

Name	ERROR_TRACTION_OPTION_1_NOT_CFG		
Description	Traction Option 1 not present Traction Option 1 requested by ATO while not present in TCMS		
Type	*Minimum Value*	*Maximum Value*	*Resolution/ Formula*
Boolean 1 bit			
Special/ Reserved Values	0	Traction Option 1 present	
	1	Traction Option 1 not present	

4.5.63 ERROR_TRACTION_OPTION_2_NOT_CFG

Name	ERROR_TRACTION_OPTION_2_NOT_CFG		
Description	Traction Option 2 not present Traction Option 2 requested by ATO while not present in TCMS		
Type	*Minimum Value*	*Maximum Value*	*Resolution/ Formula*
Boolean 1 bit			
Special/ Reserved Values	0	Traction Option 2 present	
	1	Traction Option 2 not present	

4.5.64 ERROR_TRACTIONREQUEST_NOT_CFG

Name	ERROR_TRACTIONREQUEST_NOT_CFG		
Description	Traction Request not present Traction Request request from ATO while not present in TCMS		
Type	*Minimum Value*	*Maximum Value*	*Resolution/ Formula*
Boolean 1 bit			
Special/ Reserved Values	0	Traction Request present	
	1	Traction Request not present	

4.5.65 HOLDING_BRAKE_APPLIED_TCMS

Name	HOLDING_BRAKE_APPLIED_TCMS
Description	Holding brake applied

Name	**HOLDING_BRAKE_APPLIED_TCMS**		
	Feedback signal - the vehicle braked by holding brake.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Holding brake not applied	
	1	Holding brake applied	

4.5.65.1 The use of these signals is project specific. If the holding brake is controlled completely from TCMS, these signals are not used.

4.5.65.2 Note: The *Holding brake request* signal is set when the train speed sinks below project-specific small value. The signal is reset when ATO disengages, but not earlier than 1 second after it was set (to be included in SS-125).

4.5.65.3 When TCMS detects the rising edge of *Holding brake request* signal, it applies the Holding brake (exported constraint).

4.5.65.4 If the TCMS cannot fulfil 4.5.65.3, then the FVA is responsible for this mapping.

4.5.65.5 Note: If Holding brake is requested by *Holding brake request* signal and not confirmed by *Holding brake applied* signal then service (indirect) brake shall be used by ATO-OB instead, after project-specific time delay would elapse (to be included in SS-125).

4.5.65.6 The release of Holding brake shall be done by TCMS according to its internal functions after TCMS’s internal request on tractioning appears (regardless which is the source of this traction request - ATO-OB or Driver) (exported constraint).

4.5.66 HOLDING_BRAKE_REQUEST_ACPU

Name	**HOLDING_BRAKE_REQUEST_ACPU**		
Description	Holding brake request Control signal for applying of Holding brake.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not engage the holding brake	
	1	Enage the holding brake	

4.5.67 INDIRECT_BRAKE_REQUEST_ACPU

Name	**INDIRECT_BRAKE_REQUEST_ACPU**		
Description	Immediate indirect air brake request Auxiliary control signal for direct control of indirect (train) air brake Range: 0.0 % to 100.0 %. Resolution <= 0.1% Note: 0% of brake force typically equals a brake pipe pressure of 5.0 bar, 100% equals a brake pipe pressure of 3.5 bar		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 11 bit	-100.0% -1000	100.0% 1000	0,1%

4.5.67.1 Note: As this signal enables the instant control of air brake, they also allow the forced use of air brake according to national rules.

4.5.67.2 Note: If the Direct brake is requested (by *Immediate direct air brake request*) and not confirmed by the *Direct brake applied* signal, then ATO-OB will request service (indirect) brake instead (to be included in SS-125).

4.5.68 L_PACKET

Name	L_PACKET
Description	L_PACKET indicates the length of the packet in bits, including all bits of the packet header L_PACKET is based on [24] 7.5.1.49

4.5.69 MAX_AVAILABLE_DYNAMICBRAKE_EFFORT_TCMS

Name	**MAX_AVAILABLE_DYNAMICBRAKE_EFFORT_TCMS**		
Description	Maximum available dynamic brake effort (for the whole train) Includes both multiple traction and reduced dynamic brake capabilities (isolated bogie etc.) Range: 0 ... 3000 kN, resolution 1 kN. Mandatory for Locos, optional for EMUs. The value is used for calculating the speed profiles and for country-specific limitation of EDB force. Only if Q_Max_Available_DynamicBrake_Effort = 1		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 12 bit	0 kN 0	3000 kN 3000	1 kN

4.5.70 MAX_AVAILABLE_DYNAMICBRAKE_POWER_TCMS

Name	**MAX_AVAILABLE_DYNAMICBRAKE_POWER_TCMS**		
Description	Maximum available dynamic brake power (for the whole train Includes both multiple traction and reduced dynamic brake capabilities (isolated bogie etc.) Range: 0 ... 32 000 kW, resolution 1 kW, The value is used for calculating the speed profiles. Only if Q_Max_DynamicBrake_Power = 1		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 15 bit	0 kW 0	32 000 kW 32000	1 kW

4.5.71 MAX_AVAILABLE_TRACTIVE_EFFORT_TCMS

Name	**MAX_AVAILABLE_TRACTIVE_EFFORT_TCMS**		
Description	Maximum available tractive effort (for the whole train) Includes both multiple traction and reduced traction capabilities (isolated bogie etc.) Range: 0 ... 3000 kN, resolution 1 kN The value is used for calculation of speed profiles. Only if Q_Max_Available_Traction_Effort = 1		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 12 bit	0 kN 0	3000 kN 3000	1 kN

4.5.72 MAX_AVAILABLE_TRACTIVE_POWER_TCMS

Name	**MAX_AVAILABLE_TRACTIVE_POWER_TCMS**		
Description	Maximum available tractive output power (for the whole train) Includes both multiple traction and reduced traction capabilities (isolated bogie etc.) Range: 0 ... 32 000 kW, resolution 1 kW. The value is used for calculation of speed profiles. Only if Q_Max_Available_Traction_Power = 1		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 15 bit	0 kW 0	32 000 kW 32000	1 kW

4.5.73 MAX_TRAIN_SPEED

Name	**MAX_TRAIN_SPEED**		
Description	Maximum speed of the train		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bit	0 km/h 0	600 km/h 120	5 km/h
Default value	0		

4.5.74 N_ITER

Name	**N_ITER**
Description	Number of iterations of a data set following this variable in a packet If N_ITER is 0 then no data set is following. Two nested levels of iterations can exist. N_ITER is defined in [24] 7.5.1.80

4.5.75 NID_PACKET

Name	**NID_PACKET**
Description	Packet identifier This is used in the header for each packet, allowing the receiving equipment to identify the data that follows. N_ITER is defined in [24] 7.5.1.93

4.5.76 Q_AVAILABLE_DYNAMICBRAKE_EFFORT_TCMS

Name	**Q_AVAILABLE_DYNAMICBRAKE_EFFORT_TCMS**		
Description	Qualifier for currently available dynamic brake power This flag is true when the currently available dynamic brake power is known.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Currently available dynamic brake power unknown	
	1	Currently available dynamic brake power known	

4.5.77 Q_AVAILABLE_TRACTIVE_EFFORT_TCMS

Name	**Q_AVAILABLE_TRACTIVE_EFFORT_TCMS**
Description	Qualifier for currently available tractive effort This flag is true when the currently available tractive effort is known.

Name	**Q_AVAILABLE_TRACTIVE_EFFORT_TCMS**		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Currently available tractive effort unknown	
	1	Currently available tractive effort known	

4.5.78 Q_BRAKE_MODEL_TCMS

Name	**Q_BRAKE_MODEL_TCMS**		
Description	This flag indicates if a brake model is contained in packet 33		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	no brake model available	
	1	Brake model available	

4.5.79 Q_MAX_AVAILABLE_DYNAMICBRAKE_EFFORT_TCMS

Name	**Q_MAX_AVAILABLE_DYNAMICBRAKE_EFFORT_TCMS**		
Description	Qualifier for maximum available dynamic brake effort This flag is true when the maximum available dynamic brake effort (for the whole train) is known.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Maximum available dynamic brake effort unknown	
	1	Maximum available dynamic brake effort known	

4.5.80 Q_MAX_DYNAMICBRAKE_POWER_TCMS

Name	**Q_MAX_AVAILABLE_DYNAMICBRAKE_POWER_TCMS**		
Description	Qualifier for maximum available dynamic brake power This flag is true when the maximum available dynamic brake power (for the whole train) is known.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Maximum available dynamic brake power unknown	
	1	Maximum available dynamic brake power known	

4.5.81 Q_MAX_AVAILABLE_TRACTIVE_EFFORT_TCMS

Name	**Q_MAX_AVAILABLE_TRACTIVE_EFFORT_TCMS**		
Description	Qualifier for maximum available tractive effort This flag is true when the maximum available tractive effort (for the whole train) is known.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Maximum available tractive effort unknown	

Name	Q_MAX_AVAILABLE_TRACTIVE_EFFORT_TCMS		
	1	Maximum available tractive effort known	

4.5.82 Q_MAX_AVAILABLE_TRACTIVE_POWER_TCMS

Name	Q_MAX_AVAILABLE_TRACTIVE_POWER_TCMS		
Description	Qualifier for maximum available tractive power This flag is true when the maximum available tractive power (for the whole train) is known.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Maximum available tractive power unknown	
	1	Maximum available tractive power known	

4.5.83 Q_TRAIN_MASS_TCMS

Name	Q_TRAIN_MASS_TCMS		
Description	Qualifier for train mass This flag is true train massis known.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Train mass unknown	
	1	Train mass known	

4.5.84 RELATIVE_TRACTION_REQUEST_ACPU

Name	RELATIVE_TRACTION_REQUEST_ACPU		
Description	Percentage of traction/brake capability of the train. Range: -100% (full brake) ... 0 ... +100% (full traction), resolution ≤ 0.1% In order to achieve the required precision with integer value, this value is coded with a scaling factor of 10: -100.0% is coded as -1000 100.0% is coded as 1000		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 11 bit	-100.0% -1000	100.0% 1000	0,1%

4.5.84.1 Relative traction/brake request - interpretation option I:

The *Relative traction/brake request* signal is (in positive values) defined as a percentage of **actual current traction capability** of the vehicle (see Figure 1).
Note: this definition ensures that ATO-OB shall never request an unreachable value (like requesting Ft_{max} at V_{max} or P_{max} at zero speed).
Exported constraint: At each situation, the change of this signal shall have an immediate response in TCMS - this should be understood as there will be no ineffective change (“dead

travel") of this signal with no response on TCMS's internal control signal value. Necessary times for switching the traction circuits to traction / brake schemes etc. are accepted.

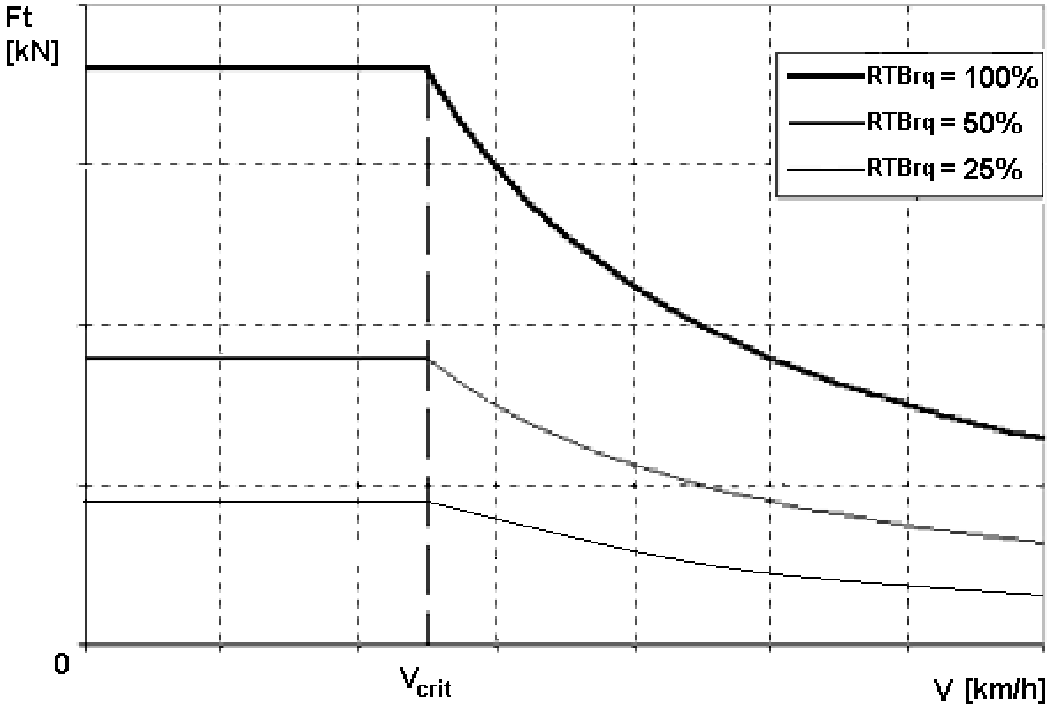


Figure 3: Relative traction/brake request - interpretation option I

4.5.84.2 **Relative traction/brake request - interpretation option II**:
In contrast to Option I, this interpretation of the requested value is related to a **speed independent maximum (reference) force** value. Here, the requested value is the percentage of the maximum available tractive / dynamic brake effort (Table 7 / Fig. 2).

For EMUs:

- the weight compensation shall be performed by the vehicle itself by varying the maximum reference force input to ATO-OB, e.g. lowering the value if the EMU is empty. As a result, for all weights the same requested percentage value requested by ATO-OB shall lead to the same kinematic acceleration/deceleration (excluding all forces external to the train);

- the maximum reference force varies only over the current load weight;
- the weight compensation on ED brake force is optional.

For both EMUs and Locomotives:

- The TCMS shall calculate the requested force applied to the vehicle as the product of the maximum reference force multiplied by the percentage value as commanded by the ATO-OB but limited with the current available speed dependent force value.

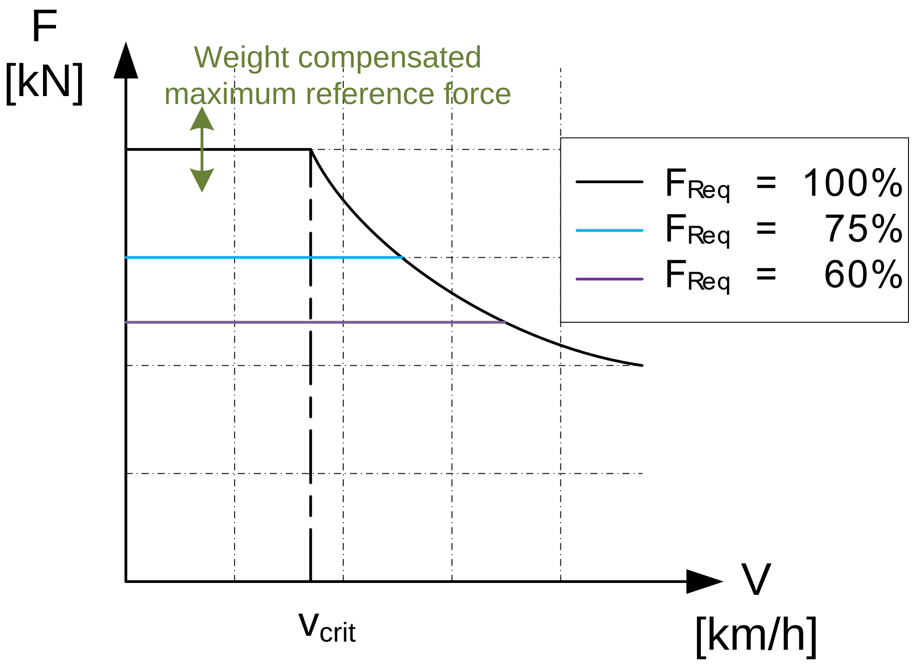


Figure 4: Relative traction/brake request - interpretation option II

4.5.84.3 Relative traction/brake request interpretation in negative values depends on the way of brake control:

- if the distribution of braking effort between dynamic and air brake is managed by TCMS itself (typically, EMUs), then the -100% request shall be interpreted as a request on full service brake (for both Option I and Option II), whereby “full service brake” here is referencing the maximum braking force;
- if this distribution is managed by ATO-OB (typically, locomotives), then the -100% request shall be interpreted as a request on full dynamic brake over the whole train. Then, chapters

5.1.2.13 and 5.1.2.14, including Figures 1 and 2, shall be used accordingly for definition of dynamic brake control.

4.5.84.4 The conversion of *Relative traction/brake request* signal to vehicle-specific control signals is a task for TCMS (exported constraint). If the TCMS is unable to do so, then the ATO shall take over this task, using the provided braking models.

4.5.84.5 Relative traction / brake request is equivalent to UIC 556 signal Traction target value: telegram R1, octet 49 + 50, signal 4.23/1

4.5.84.6 Relative traction / brake request is a mandatory signal. It is processed by the functional vehicle interface.

4.5.84.7 The TCMS uses this information to realize the ATO-OB request on traction / brake capabilities of the train.

4.5.85 RELEASE_QUICK_BRAKE_ACPU

Name	**RELEASE_QUICK_BRAKE_ACPU**		
Description	Quick brake release request Auxiliary signal for quick brake release (mandatory for Locos, optional for EMUs). The function will be handled in TCMS using Low pressure overfilling (Angleicher) and/or High pressure filling stroke (Füllstoss)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not release the quick brake	
	1	Release the quick brake	

4.5.86 SB_APPLIED_TCMS

Name	**SB_APPLIED_TCMS**		
Description	Service Brake applied Service brake applied (pressure at brake distributor output >= project specific small value)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	SB not applied	
	1	SB applied	

4.5.86.1 The *SB applied* signals are mandatory for both Locos and EMUs.

4.5.86.2 Note: The ATO-OB uses the *SB applied* information to block the positive value of *Relative traction/brake request* output signal (to be included in SS-125).

4.5.86.3 *SB applied* signal should always be set when a service brake (triggered by ATO-OB or Driver) has been applied (at least, at minimum applicable level) and shall be reset after complete brake release (exported constraint).

4.5.86.4 If Emergency brake is applied, then *SB applied* signal is set as well (exported constraint).

4.5.86.5 If 4.5.86.3 cannot be fulfilled by the TCMS, then the FVA is responsible for setting the *SB applied* signal.

4.5.87 SPEED_SENSOR_STATUS_TCMS

Name	**SPEED_SENSOR_STATUS_TCMS**		
Description	Speed sensors status Per axle		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned integer 4 bit			
Special/ Reserved Values	0	Speed sensors status disabled	
	1	Speed sensors status OK	
	3	Spare	
	4	Speed sensors status Error	
	5-7	Spare	

4.5.87.1 Note: The speed sensor status variables is referring to one single axle. Data from multiple axles can be handled at packet level (iterated values)

4.5.88 SPEED_SENSOR_PULSES_TCMS

Name	**SPEED_SENSOR_PULSES_TCMS**		
Description	Pulser per km of wheelspeed sensor		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned integer 20 bit	10 000 10 000	1 000 000 1 000 000	1 pulse
Special/ Reserved Values	0	No information	
	1	Sensor failure	
	2-9999	spare	

4.5.89 TB_LEVER_FAILURE_TCMS

Name	**TB_LEVER_FAILURE_TCMS**		
Description	T/B lever failure This flag is true when the T/B lever position is unknown (T/B Lever failure)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	No T/B lever failure	
	1	T/B lever failure	

4.5.90 TB_LEVER_TCMS

Name	**TB_LEVER_TCMS**
Description	T/B lever position Indication of traction / zero / brake position of TBL

Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	TBL Zero	
	1	TBL Traction	
	2	TBL Brake	
	3	Spare	

4.5.91 TB_SET_TCMS

Name	TB_SET_TCMS		
Description	T/B set value Current value of TCMS's traction/brake control signal ATO-OB uses this information for smooth Man → Aut transition Expressed in kN		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 13 bit	-3000 kN -3000	3000 kN 3000	1 kN

4.5.92 TCMS_CAPABILITIES_REQUEST_ACPU

Name	TCMS_CAPABILITIES_REQUEST_ACPU		
Description	TCMS capabilities request Request for information about TCMS capabilities. Note: the functional vehicle adaptor must be configured accordingly		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	No TCMS capabilities packet requested	
	1	TCMS capabilities packet requested	

4.5.92.1 The signal TCMS capabilities request is mandatory.

4.5.92.2 The signal TCMS capabilities request is sporadic

4.5.92.3 The signal TCMS capabilities request must be sent by the ATO at system start up.

4.5.93 TIME_OFFSET_MS_TCMS

Name	TIME_OFFSET_MS_TCMS		
Description	Absolute onboard time offset, expressed in UNIX time format ms component of total time The fractional part of the offset between ATO Master time and TCMS Master time. Note: This number is always positive		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 32 bits	0 ms 0 ms	999 ms 999	1 ms

4.5.94 TIME_OFFSET_SIGN_TCMS

Name	TIME_OFFSET_SIGN_TCMS		
Description	Qualifier, determines if ATO master clock value is smaller or larger than the TCMS master clock		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 3 bit			
Special/ Reserved Values	0	Offset unknown	

	1	No offset	
	2	ATO time > TCMS time	
	3	ATO time < TCMS time	
	4-7	Spare	

4.5.95 TIME_OFFSET_TCMS

Name	**TIME_OFFSET_TCMS**		
Description	ATO onboard time, expressed in UNIX time format Unsigned integer shall be used,		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned integer 32 bits	0 s 0	2147483647 s 2147483647	1 s

4.5.96 TRACTION_APPLIED_TCMS

Name	**TRACTION_APPLIED_TCMS**		
Description	Traction applied Explanation: Propulsion reports that traction is applied		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Traction not applied	
	1	Traction applied	

4.5.97 TRACTION_OVER_BRAKE_ENABLED_TCMS

Name	**TRACTION_OVER_BRAKE_ENABLED_TCMS**		
Description	Traction over brake enabled TCMS informs ATO-OB about fact that it is possible to request traction even if service brake is applied. This signal covers for example brake cleaning mode or hill start.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Traction over brake not enabled	
	1	Traction over brake enabled	

4.5.97.1 The *Traction over brake enabled* signal is mandatory if a function needing such signal is included in TCMS.

4.5.97.2 Note: The *Traction over brake enabled* information is used by ATO-OB to limit the positive value of *Relative traction/brake request* according to project-specific parameters (in time and/or value of *Relative traction/brake request*). This enables ATO-OB to request limited traction in specific situations (brake cleaning mode, hill start) even in the case when the service brake is applied (to be included in SS-125).

4.5.97.3 Note: In most cases, the TCMS will rely on a driver data entry to distinguish between the different situations (brake cleaning mode, hill start).

4.5.97.4 If *Traction over brake enabled* signal is set, then TCMS shall not send *SB applied* signal, if no other request on service brake is active (exported constraint).

4.5.97.5 If 4.5.97.4 cannot be fulfilled by the TCMS, then the FVA is responsible for fulfilling the conditions related to the *SB applied* signal.

4.5.98 TRACTION_READY_TCMS

Name	**TRACTION_READY_TCMS**		
Description	Traction ready All conditions for applying the traction are fulfilled (propulsion ready, etc.). If this signal disappears during the run, ATO keeps engaged, but it sets coasting. When the signal re-appears, traction can be applied automatically.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Traction not ready	
	1	Traction ready	

4.5.98.1 If TCMS requests to confirm some situation by “Forced zero” (in manual driving: by setting the TBL to zero position), it will reset the *Traction ready* signal and ATO-OB limits its positive output to zero, until *Traction ready* signal re-appears. If TCMS needs driver’s confirmation, this cannot be done by setting TBL to zero (as TBL already is there) and other solution must be found in TCMS (exported constraint).

4.5.99 TRACTION_REQUEST_ACPU

Name	**TRACTION_REQUEST_ACPU**		
Description	Auxiliary control signal for traction control		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	No Traction requested	
	1	Traction requested	

4.5.99.1 Traction request is a mandatory signal. It is processed by the functional vehicle interface.

4.5.99.2 Traction request is equivalent to UIC 556 signals Prepare for running, Prepare for braking: telegram R1, octet 48, bits 2 + 3, signal 4.34/1

4.5.100 TRAIN_MASS_TCMS

Name	**TRAIN_MASS_TCMS**		
Description	Train mass		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 14 bit	0 t 0	15000 t 15000	1 t

4.5.101 TRAVELLED_DISTANCE_TCMS

Name	**TRAVELLED_DISTANCE_TCMS**		
Description	Travelled distance TCMS's odometry counter (ATO format) Range: -2^{31} … 0 … $+(2^{31} - 1)$ mm, resolution 1 mm (max: +/- 2 147 km)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 32 bit	-2^{31} mm	2^{31} mm	1 mm

4.5.101.1 ATO-OB uses distance information for on-track localisation, computing speed profiles and for train control.

4.5.101.2 At least, the *Travelled distance* signal must be stamped with time stamp (of TCMS's board clock, accuracy <= 1 ms) when this signal was processed by TCMS. Next, the packet containing this signal must be stamped with time stamp when it was transmitted (or, taken for transmitting) (exported constraint).

4.5.101.3 Travelled *distance* signal is incremented when the vehicle moves in direction of active cabin and is decremented when it is moving in opposite direction.

4.5.102 TSI_STANDSTILL_TCMS

Name	**TSI_STANDSTILL_TCMS**		
Description	TSI standstill Logical information about standstill according to TSI		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Standstill not reached	
	1	TSI Standstill reached	

4.5.102.1 *TSI standstill* information is used for functions related to standstill (for example holding brake control, door control etc.)

4.5.103 UTC_MASTER_TCMS

Name	**UTC_MASTER_TCMS**		
Description	Configuration of master time		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	UTM_TCMS_only	Only TCMS Time available
	1	UTM_TCMS_master	TCMS and ATO time available, TCMS is master
	2	UTM_ATO_master	TCMS and ATO time available, ATO is master

4.5.103.1 Note: See 5.5.1, Reference Time for time management.

4.5.104 UTC_TIME_ACPU

Name	**UTC_TIME_ACPU**		
Description	ATO onboard time, expressed in UNIX time format Unsigned integer shall be used,		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned integer 32 bits	0 s 0	2147483647 s 2147483647	1 s

4.5.104.1 Unsigned integer shall be used,

4.5.104.1.1Note: Unsigned integer will avoid the wrapover on 19. Jan 2038

4.5.104.1.2Note: Most POSIX standard libraries utilize 32-bit signed

4.5.104.2 Note: See 5.5.1, Reference Time for time management.

4.5.105 UTC_TIME_MS_ACPU

Name	**UTC_TIME_MS_ACPU**		
Description	ATO onboard time, expressed in UNIX time format ms component of total time		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned integer 32 bits	0 ms 0	999 ms 999	1 ms

4.5.105.1 32 bits are required in order to ensure compatibility with the related variables

4.5.106 UTC_TIME_MS_TCMS

Name	**UTC_TIME_MS_TCMS**		
Description	TCMS onboard time, expressed in UNIX time format ms component of total time		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned integer 32 bits	0 ms 0	999 ms 999	1 ms

4.5.106.1 32 bits are selected in order to ensure compatibility with the related variables

4.5.107 UTC_TIME_TCMS

Name	**UTC_TIME_TCMS**		
Description	TCMS onboard time, expressed in UNIX time format Unsigned integer shall be used,		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned integer 32 bits	0 s 0	2147483647 s 2147483647	1 s
Special/ Reserved Values	MSB is spare		

4.5.107.1 Unsigned integer shall be used,

4.5.107.1.1Note: Unsigned integer will avoid the wrapover on 19. Jan 2038

4.5.107.1.2Note: Most POSIX standard libraries utilize 32-bit signed

4.5.108 WHEEL_DIAMETER_TCMS

Name	**WHEEL_DIAMETER_TCMS**		
Description	Current value of wheel diameters. Range: 300 … 2000 mm, resolution 0,1 mm. Special value for "not used". The variable is used if odometry is processed by ATO-OB from raw sensor signals. Note: ATO-OB uses this information for its own odometry.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 16 bit	300 mm 3000	2000 mm 20000	1 mm

5 ATO Functional Vehicle Adapter

5.1 General

5.1.1.1 The Functional Vehicle Adapter (FVA) encapsulates all vehicle specific information in a way that allows plug & play replacement of the ATO.

5.1.1.2 Note: The combination of encapsulation of the project- specific data and the definition of precise message sequences define, in combination, a message-level API.

5.2 Parameters

5.2.1 P_ACTUAL_INPUTCURRENT

Name	**P_ACTUAL_INPUTCURRENT**		
Description	Actual input current Actual value of input current (for the whole train) Range: - 10 000 A... 0 ... + 10 000 A, resolution <= 1 A (10 A @ DC systems) (negative values refer to regenerative brake current) Note: ATO-OB uses this variable for maintaining the track condition “limitation of input current”. Note: Negative values (regenerative braking) are mandatory for Locos, for EMUs they are optional. This parameter provides (optionally) a static value for this variable		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 16 bit	-10000 -10000	10000 10000	1
Default value	0		

5.2.2 P_ACTUAL_INPUTCURRENT_CFG

Name	**P_ACTUAL_INPUTCURRENT_CFG**		
Description	Actual input current information implemention FVA configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 3 bit			
Special/ Reserved Values	0	No actual input current information available	
	1	Actual input current information implemented dynamically through FVA	
	2	Actual input current information implemented dynamically through TCMS	
	3	Actual input current information implemented statically through FVA	
	4-7	Spare	
Default value	0: No actual input current information available		

5.2.3 P_AdhesionFactor_Reduction

Name	**P_ADHESIONFACTOR_REDUCTION**
Description	Adhesion factor reduction Reduction of adhesion (for informing ATO-TS)

	Values 0 and 1 for reporting the bad adhesion according to ETCS convention. Values: 10 (really bad adhesion) … 100 % (full adhesion, no limitation), are reserved for future use.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 8 bit	0	100	1
Default value	0		

5.2.4 P_ADHESIONFACTOR_REDUCTION_CFG

Name	**P_ADHESIONFACTOR_REDUCTION_CFG**		
Description	Adhesion factor reduction value FVA configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 3 bit			
Special/ Reserved Values	0	No adhesion factor reduction value available	
	1	Adhesion factor reduction value implemented dynamically through FVA	
	2	Adhesion factor reduction value implemented dynamically through TCMS	
	3	Adhesion factor reduction value implemented statically through FVA	
	4-7	Spare	
Default value	0		

5.2.5 P_AFB_SPEED_INSTALLED

Name	**P_AFB_SPEED_INSTALLED**		
Description	AFB (speed setting) installed TCMS is „Automatischer Fahrbetrieb“ capable (speed preset)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	No AFB (speed setting) installed	
	1	AFB (speed setting) installed	
Default value	0 No AFB (speed setting) installed		

5.2.6 P_AFB_TRACTION_INSTALLED

Name	**P_AFB_TRACTION_INSTALLED**		
Description	AFB (traction setting) installed TCMS is „Automatischer Fahrbetrieb“ capable (traction preset)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	No AFB (traction setting) installed	
	1	AFB (traction setting) installed	
Default value	0 No AFB (traction setting) installed		

5.2.7 P_ATO_DIRECT_BRAKE_CONTROL

Name	**P_ATO_DIRECT_BRAKE_CONTROL**		
Description	Set if the ATO shall control the direct brake directly		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	No direct brake control by the ATO	
	1	Direct brake control by the ATO	
Default value	0 No direct brake control by the ATO		

5.2.8 P_ATO_HOLDING_BRAKE_CONTROL

Name	**P_ATO_HOLDING_BRAKE_CONTROL**		
Description	Set if the ATO shall control the holding brake directly		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	No holding brake control by the ATO	
	1	Holding brake control by the ATO	
Default value	0 No holding brake control by the ATO		

5.2.9 P_BRAKE_BLENDING_INSTALLED

Name	**P_BRAKE_BLENDING_INSTALLED**		
Description	Brake Blending installed TCMS is capable of brake blending		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	No Brake Blending installed	
	1	Brake Blending installed	
Default value	0 No Brake Blending installed		

5.2.10 P_BRAKE_MODE_CFG

Name	**P_BRAKE_MODE_CFG**		
Description	Brake mode implementation		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No Brake mode information present	
	1	Brake mode configuration available via FVA	
	2	Brake mode configuration available from TCMS	
Default value	0 No Brake mode information present		

5.2.11 P_BRAKE_MODEL_CFG

Name	**P_BRAKE_MODEL_CFG**		
Description	Brake model present A brake model is available (from TCMS or from the Functional Vehicle Adaptor)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No Brake model present	

Name	**P_BRAKE_MODEL_CFG**		
	1	Brake model present at FVA	
	2	Brake model present at TCMS	
Default value	0 No Brake model present		

5.2.12 P_BRAKE_MODELS

Parameter Name	**P_BRAKE_MODELS**		
Description	Model of the emergency brake, traction, and service brake (if present), to be used by the Core CPU		
Content	**Variable**	**Length**	**Comment**
	N_ITER	5	0 …9
	DECELERATION_CLASS_ID	8	
	BRAKE_DELAY_CLASS_ID	8	
	DMOD_MODEL_BEGIN_BRAKE_TCMS	8	Part of EB model
	DMOD_MODEL_FULL_BRAKE_TCMS	11	Part of EB model
	N_ITER	5	Part of EB model max. value: 5
	DMOD_MODEL_SPEED_TCMS(k)	8	Part of EB model
	DMOD_MODEL_DECELER_TCMS(k)	8	Part of EB model
	DMOD_CUT_TRACT_DELAY_TCMS	8	Part of traction model
	DMOD_TRAIN_MAX_ACC_TCMS	10	Part of traction model
	DMOD_ACC_COEF_SB_UNUSED_TCMS	2	Part of traction model
	DMOD_ACC_COEF_SB_USED_TCMS	2	Part of traction model
	Q_SB_MODEL_cfg	1	Part of SB model
	DMOD_MODEL_BEGIN_BRAKE_TCMS	8	Part of SB model
	DMOD_MODEL_FULL_BRAKE_TCMS	11	Part of SB model
	N_ITER	5	Part of SB model max. value: 5
	DMOD_MODEL_SPEED_TCMS(k)	8	Part of SB model
	DMOD_MODEL_DECELER_TCMS(k)	8	Part of SB model
	DMOD_MIN_ROT_MASS_PERCENT_TCMS	8	Part of rot mass model
	DMOD_NOM_ROT_MASS_PERCENT_TCMS	8	Part of rot mass model
	DMOD_MAX_ROT_MASS_PERCENT_TCMS	8	Part of rot mass model
	DMOD_T_W_TCMS	13	Part of driver delay
	DMOD_T_P_TCMS	13	Part of driver delay
	DMOD_T_I_P_TCMS	13	Part of driver delay
	DMOD_T_RSMA_TCMS	13	Part of driver delay
Default value	All values set to 0		

5.2.12.1.1 It shall be possible to store up to 10 brake model data sets

5.2.12.2 P_N_BRAKE_MODELS

Parameter Name	P_N_BRAKE_MODELS		
Description	Determines how many Brake Models are available This parameter describes the configuration of the TCMS Interface. This parameter is project specific and persistent.		
Type	**Minimum Value**	**Maximum Value**	**Resolution/ Formula**
Integer 4 bits	0	9	1
The default value is 9. This parameter is only relevant if P_Q_BRAKE_MODELS is >0			

5.2.12.3 P_Q_BRAKE_MODELS

Parameter Name	**P_Q_BRAKE_MODELS**		
Description	Determines if Brake Models are available This parameter describes the configuration of the TCMS Interface. This parameter is project specific and persistent.		
Type	**Minimum Value**	**Maximum Value**	**Resolution/ Formula**
Integer 2 bits			

Parameter Name	**P_Q_BRAKE_MODELS**	
Special/ Reserved Values	**0**	**BM_not_present: No brake models avaliable**
	1	BM_Fixed: Fixed brake model parameters are stored in the Functional Vehicle Adaptor
	2	BM_EXT: Brake models can be received from the vehicle via external interface
	3	Spare
Note: The default setting for this parameter is highlighted in **bold** letters.		

5.2.13 P_ BRAKEREQUEST_CFG

Name	**P_BRAKEREQUEST_CFG**		
Description	Brake request present		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No Brake request available	
	1	Brake request implemented by FVA	
	2	Brake request implemented by TCMS	
Default value	0 No Brake request available		

5.2.14 P_CURRENTLY_AVAILABLE_DYNAMICBRAKE_EFFORT

Name	**P_CURRENTLY_AVAILABLE_DYNAMICBRAKE_EFFORT**		
Description	Preset value for currently available dynamic brake effort Max. dynamic brake effort at current speed. Includes both multiple traction and reduced dynamic brake capabilities (isolated bogie etc.) Range: 0 ... 3000 kN, resolution 1 kN,		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 12 bit	0 kN 0	3000 kN 3000	1 kN
Default value	0		

5.2.15 P_CURRENTLY_AVAILABLE_DYNAMICBRAKE_CFG

Name	**P_CURRENTLY_AVAILABLE_DYNAMICBRAKE_EFFORT_CFG**		
Description	Currently available dynamic brake effort present FVA configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 3 bit			
Special/ Reserved Values	0	No currently available dynamic brake effort available	
	1	Currently available dynamic brake effort implemented dynamically through FVA	
	2	Currently available dynamic brake effort implemented dynamically through TCMS	
	3	Currently available dynamic brake effort implemented statically through FVA	
	4-7	Spare	
Default value	0 No currently available dynamic brake effort available		

5.2.16 P_CURRENTLY_AVAILABLE_TRACTIVE_EFFORT

Name	**P_CURRENTLY_AVAILABLE_TRACTIVE_EFFORT**		
Description	Static value Currently available tractive effort (for the whole train) Includes both multiple traction and reduced traction capabilities (isolated bogie etc.) Range: 0 ... 3000 kN, resolution 1 kN The value is used for calculation of speed profiles.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 12 bit	0 kN 0	3000 kN 3000	1 kN
Default value	0		

5.2.17 P_CURRENTLY_AVAILABLE_TRACTIVE_EFFORT_CFG

Name	**P_CURRENTLY_AVAILABLE_TRACTIVE_EFFORT_CFG**		
Description	Currently available tractive power present FVA configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 3 bit			
Special/ Reserved Values	0	No Currently available tractive power available	
	1	Currently available tractive power implemented dynamically through FVA	
	2	Currently available tractive power implemented dynamically through TCMS	
	3	Currently available tractive power implemented statically through FVA	
	4-7	Spare	
Default value	0 No Currently available tractive power available		

5.2.18 P_DMOD_ACC_COEF_SB_UNUSED_TCMS

Name	**P_DMOD_ACC_COEF_SB_UNUSED_TCMS**		
Description	Acceleration coefficient when the service brake is not present or not available. Ponderation coefficient to be applied on maximum train acceleration when the service brake is not available.		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 7 bits	0 0	1 100	0,01
Default value	0		

5.2.19 P_DMOD_ACC_COEF_SB_USED_TCMS

Name	**P_DMOD_ACC_COEF_SB_USED_TCMS**		
Description	Acceleration coefficient when the service brake is available. Ponderation coefficient to be applied on maximum train acceleration acceleration when the service brake is available.		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 7 bits	0 0	1 100	0,01
Default value	0		

5.2.20 P_DMOD_MODEL_BEGIN_BRAKE_TCMS

Name	**P_DMOD_MODEL_BEGIN_BRAKE_TCMS**		
Description	Delay between ordering a brake application, and when brake begins to be applied (more than 0%)		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bit	0 s 0	25,5 s 255	0,1 s
Default value	0		

5.2.21 P_DMOD_CUT_TRACT_DELAY_TCMS

Name	**P_DMOD_CUT_TRACT_DELAY_TCMS**		
Description	Delay to cut off traction Delay between the ordering of traction cut off and the effective cut off of the traction		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bits	0 s 0	25,5 s 255	0,1 s
Default value	0		

5.2.22 P_DMOD_MODEL_DECELER_TCMS

Name	**P_DMOD_MODEL_DECELER_TCMS**		
Description	Brake model deceleration point Coordinate on the Y axis (=train deceleration) of a point of the deceleration model		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bit	0 m/s2 0	25,5 m/s2 255	0,1 m/s2
Default value	0		

5.2.23 P_DMOD_MODEL_FULL_BRAKE_TCMS

Name	**P_DMOD_MODEL_FULL_BRAKE_TCMS**		
Description	Delay between when the braking effort begins (>0%) and when the full braking effort is reached (100%)		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 11 bits	0 s 0	120, 0 s 1200	0,1 s
Default value	0		

5.2.24 P_DMOD_TRAIN_MAX_ACC_TCMS

Name	**P_DMOD_TRAIN_MAX_ACC_TCMS**		
Description	Maximum acceleration that the train is able to reach		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 10 bits	0 m/s^2 0	10,23 m/s^2 1023	0,01 m/s^2
Default value	0		

5.2.25 P_DMOD_MAX_ROT_MASS_PERCENT_TCMS

Name	**P_DMOD_MAX_ROT_MASS_PERCENT_TCMS**		
Description	maximum rotating mass percentage maximum rotating mass of the train, expressed as a percentage of the total weight of the train		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bit	0% 0	25,5 % 255	0,1 %

Name	P_DMOD_MAX_ROT_MASS_PERCENT_TCMS
Default value	0

5.2.26 P_DMOD_MIN_ROT_MASS_PERCENT_TCMS

Name	P_DMOD_MIN_ROT_MASS_PERCENT_TCMS		
Description	minimum rotating mass percentage minimum rotating mass of the train, expressed as a percentage of the total weight of the train		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bit	0 % 0	25,5 % 255	0,1 %
Default value	0		

5.2.27 P_DMOD_MODEL_SPEED_TCMS

Name	P_DMOD_MODEL_SPEED_TCMS		
Description	Brake model speed point Coordinate on the X axis (=train speed) of a point of the deceleration model		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bit	0 km/h 0	600 km/h 120	5 km/h
Default value	0		

5.2.28 P_DMOD_NOM_ROT_MASS_PERCENT_TCMS

Name	P_DMOD_NOM_ROT_MASS_PERCENT_TCMS		
Description	nominal rotating mass of the train, expressed as a percentage of the total weight of the train		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 8 bit	0% 0	25,5 % 255	0,1 %
Default value	0		

5.2.29 P_DMOD_T_P_TCMS

Name	P_DMOD_T_P_TCMS		
Description	T_p parameter used by the TCMS in the braking curve calculation		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 13 bits	0 s 0	600 s 6000	0,1s
Default value	0		

5.2.30 P_DMOD_T_I_P_TCMS

Name	P_DMOD_T_I_P_TCMS		
Description	T_i_p parameter used by the TCMS in the braking curve calculation		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 13 bits	0 s 0	600 s 6000	0,1s
Default value	0		

5.2.31 P_DMOD_T_W_TCMS

Name	**P_DMOD_T_W_TCMS**		
Description	T_w parameter used by the TCMS in the braking curve calculation		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 13 bits	0 s 0	600 s 6000	0,1s
Default value	0		

5.2.32 P_DMOD_T_RSMA_TCMS

Name	**P_DMOD_T_RSMA_TCMS**		
Description	T_rsma parameter used by the TCMS in the braking curve calculation		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 13 bits	0 s 0	600 s 6000	0,1s
Default value	0		

5.2.33 P_DOORENABLE_CFG

Name	**P_DOORENABLE_CFG**		
Description	Door Enable configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No Door Enable function installed	
	1	Door Enable function implemented through FVA	
	2	Door Enable function implemented through TCMS	
Default value	0 No Door Enable function installed		

5.2.34 P_DYNAMIC_BRAKE_ENABLED

Name	**P_DYNAMIC_BRAKE_ENABLED**		
Description	Dynamic brake enabled To be set to true if the Dynamic Brake is enabled		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Dynamic brake disabled	
	1	Dynamic brake enabled	
Default value	0 Dynamic brake disabled		

5.2.35 P_DYNAMICBRAKE_CFG

Name	**P_DYNAMICBRAKE_CFG**		
Description	Dynamic brake configration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No dynamic brake installed	
	1	Dynamic brake implented through FVA	
	2	Dynamic brake implemented through TCMS	

Default value	0 No dynamic brake installed

5.2.35.1 P_STANDARD_139_CFG

Name	**P_STANDARD_139_CFG**		
Description	Capability information The TCMS is fully Subset-139 compliant		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	The TCMS is not Subset-139 compliant	
	1	The TCMS is fully Subset-139 compliant	

5.2.36 P_ENGAGEMENT_READY_cfg

Name	**P_ENGAGEMENT_READY_CFG**		
Description	Engagement ready signal present		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No Engagement ready signal installed	
	1	Engagement ready signal implented through FVA	
	2	Engagement ready signal implemented through TCMS	
Default value	0 No Engagement ready signal installed		

5.2.37 P_ERRORS

Name	**P_ERRORS**		
Description	Error messages sent to ATO or not.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	P_ERRORS_noMessage	No error messages sent to the ATO in case of configuration error
	1	P_ERRORS_Message	Error messages sent to the ATO in case of configuration error
Default value	0 No error messages sent to the ATO in case of configuration error		

5.2.38 P_HOLDING_BRAKE_CFG

Name	**P_HOLDING_BRAKE_CFG**		
Description	Holding brake installed		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No Holding brake installed	
	1	Holding brake implented through FVA	

Name	**P_HOLDING_BRAKE_CFG**		
	2	Holding brake implemented through TCMS	
Default value	0 No Holding brake installed		

5.2.39 P_MAX_AVAILABLE_DYNAMICBRAKEEFFORT

Name	**P_MAX_AVAILABLE_DYNAMICBRAKEEFFORT**		
Description	Preset value for maximum available dynamic brake effort Includes both multiple traction and reduced dynamic brake capabilities (isolated bogie etc.) Range: 0 ... 3000 kN, resolution 1 kN. Mandatory for Locos, optional for EMUs. The value is used for calculating the speed profiles and for country- specific limitation of EDB force.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 10 bit	0 kN 0	3000 kN 3000	1 kN
Default value	0		

5.2.40 P_MAX_AVAILABLE_DYNAMICBRAKEEFFORT_CFG

Name	**P_MAX_AVAILABLE_DYNAMICBRAKEEFFORT_CFG**		
Description	Maximum available dynamic brake effort configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 3 bit			
Special/ Reserved Values	0	No Maximum available dynamic brake effort available	
	1	Maximum available dynamic brake effort implemented dynamically through FVA	
	2	Maximum available dynamic brake effort implemented dynamically through TCMS	
	3	Maximum available dynamic brake effort implemented statically through FVA	
	4-7	Spare	
Default value	0 No Maximum available dynamic brake effort available		

5.2.41 P_MAX_AVAILABLE_DYNAMICBRAKE_POWER

Name	**P_MAX_AVAILABLE_DYNAMICBRAKE_POWER**		
Description	Preset value for maximum available dynamic brake power Includes both multiple traction and reduced dynamic brake capabilities (isolated bogie etc.) Range: 0 ... 32 000 kW, resolution 1 kW, The value is used for calculating the speed profiles.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 15 bit	0 kW 0	32 000 kW 32000	1 kW
Default value	0 kW		

5.2.42 P_MAX_AVAILABLE_DYNAMICBRAKE_POWER_CFG

Name	P_MAX_AVAILABLE_DYNAMICBRAKE_POWER_CFG		
Description	Maximum dynamic brake power present		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 3 bit			
Special/ Reserved Values	0	No Maximum dynamic brake power available	
	1	Maximum dynamic brake power implemented dynamically through FVA	
	2	Maximum dynamic brake power implemented dynamically through TCMS	
	3	Maximum dynamic brake power implemented statically through FVA	
	4-7	Spare	
Default value	0 No Maximum dynamic brake power available		

5.2.43 P_MaxAvailTractiveEffort

Name	P_MAXAVAILTRACTIVEEFFORT		
Description	Maximum available tractive effort (for the whole train) Includes both multiple traction and reduced traction capabilities (isolated bogie etc.) Range: 0 ... 3000 kN, resolution 1 kN The value is used for calculation of speed profiles. Only if Q_Max_Available_Traction_Effort = 1		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 10 bit	0 kN 0	3000 kN 3000	1 kN
Default value	0		

5.2.44 P_MAXAVAILTRACTIVEEFFORT_CFG

Name	P_MAX_AVAILTRACTIVEEFFORT_CFG		
Description	Maximum available tractive power configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 3 bit			
Special/ Reserved Values	0	No Maximum available tractive power available	
	1	Maximum available tractive power implemented dynamically through FVA	
	2	Maximum available tractive power implemented dynamically through TCMS	
	3	Maximum available tractive power implemented statically through FVA	
	4-7	Spare	
Default value	No Maximum available tractive power available		

5.2.45 P_MAXAVAILTRACTIVEPOWER

Name	**P_MAXAVAILTRACTIVEPOWER**		
Description	Maximum available tractive output power (for the whole train) Includes both multiple traction and reduced traction capabilities (isolated bogie etc.) Range: 0 … 32 000 kW, resolution 1 kW. The value is used for calculation of speed profiles. Only if Q_Max_Available_Traction_Power = 1		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 15 bit	0 kW 0	32 000 kW 32000	1 kW
Default value	0 kW		

5.2.46 P_MAX_AVAILTRACTIVEPOWER_CFG

Name	**P_MAX_AVAILTRACTIVEPOWER_CFG**		
Description	Maximum available traction power configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 3 bit			
Special/ Reserved Values	0	No Maximum available traction effort available	
	1	Maximum available traction effort implemented dynamically through FVA	
	2	Maximum available traction effort implemented dynamically through TCMS	
	3	Maximum available traction effort implemented statically through FVA	
	4-7	Spare	
Default value	0 No Maximum available traction effort available		

5.2.47 P_QUICKBRAKE_CFG

Name	**P_QUICKBRAKE_CFG**		
Description	Quick brake configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No Quick brake installed	
	1	Quick brake implented through FVA	
	2	Quick brake implemented through TCMS	
Default value	0 No Quick brake installed		

5.2.48 P_PNEUBRAKE_CFG

Name	**P_PNEUBRAKE_CFG**		
Description	High- level pneumatic brake control configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No High- level pneumatic brake control installed	

	1	High- level pneumatic brake control implemented through FVA	
	2	High- level pneumatic brake control implemented through TCMS	
Default value	0 No High- level pneumatic brake control installed		

5.2.49 P_REL_INDIRECTBRAKE_CFG

Name	**P_REL_INDIRECTBRAKE_CFG**		
Description	Relative Immediate Indirect Brake command configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No Relative Immediate Indirect Brake command installed	
	1	Relative Immediate Indirect Brake command implented through FVA	
	2	Relative Immediate Indirect Brake command implemented through TCMS	
Default value	0 No Relative Immediate Indirect Brake command installed		

5.2.50 P_RELTRACTIONREQUEST_CFG

Name	**P_RELTRACTIONREQUEST_CFG**		
Description	Relative Traction and Brake command configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0		No Relative Traction and Brake command installed
	1		Relative Traction and Brake command implemented through FVA
	2		Relative Traction and Brake command implemented through TCMS
Default value	0 No Relative Traction and Brake command installed		

5.2.51 P_TB_SET

Name	**P_TB_SET**		
Description	T/B set value Current value of TCMS's traction/brake control signal ATO-OB uses this information for smooth Man → Aut transition Expressed in kN		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 13 bit	-3000 kN -3000	3000 kN 3000	1 kN
Default value	0		

5.2.52 P_TB_SET_CFG

Name	**P_TB_SET_CFG**		
Description	T/B set/ preset value implementation		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 3 bit			
Special/ Reserved Values	0	No T/B set/ preset value available	
	1	T/B set/ preset value dynamically through FVA	
	2	T/B set/ preset value dynamically through TCMS	
	3	T/B set/ preset value implemented statically through FVA	
	4-7	Spare	
Default value	0 No T/B set/ preset value available		

5.2.53 P_TCMS_SB_WHEN_EB

Name	**P_TCMS_SB_WHEN_EB**		
Description	SB applied signal set by TMCS True if TCMS automatically sets SB applied signal when EB is applied.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	No SB applied signal set by TMCS	
	1	SB applied signal set by TMCS	
Default value	0 No SB applied signal set by TMCS		

5.2.54 P_TRACTIONAPPLIED_CFG

Name	**P_TRACTIONAPPLIED_CFG**		
Description	Traction applied signal configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No Traction applied signal available	
	1	Traction applied signal implemented through FVA	
	2	Traction applied signal implemented through TCMS	
Default value	0 No Traction applied signal available		

5.2.55 P_TRACTION_OPTION_1_CFG

Name	**P_TRACTION_OPTION_1_CFG**		
Description	Traction/ Brake Option 1 configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	No Traction/ Brake Option 1 present	
	1	Traction/ Brake Option 1 present	
Default value	0 No Traction/ Brake Option 1 present		

5.2.56 P_TRACTION_OPTION_2_CFG

Name	P_TRACTION_OPTION_2_CFG		
Description	Traction/ Brake Option 2 present		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	No Traction/ Brake Option 2 present	
	1	Traction/ Brake Option 2 present	
Default value	0 No Traction/ Brake Option 2 present		

5.2.57 P_TRACTION_OVER_BRAKE_CFG

Name	P_TRACTION_OVER_BRAKE_CFG		
Description	Traction over brake configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No Traction over brake	
	1	Traction over brake implemented through FVA	
	2	Traction over brake implemented through TCMS	
Default value	0 No Traction over brake		

5.2.58 P_TRACTION_READY_CFG

Name	P_TRACTION_READY_CFG		
Description	Traction ready signal configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No Traction ready signal available	
	1	Traction ready signal implemented through FVA	
	2	Traction ready signal implemented through TCMS	
Default value	0 No Traction ready signal available		

5.2.59 P_FULL_OCORA_CFG

Name	P_FULL_OCORA_CFG		
Description	Capability information The TCMS is fully OCORA compliant		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	The TCMS is not OCORA compliant	
	1	The TCMS is fully OCORA compliant	

5.2.60 P_TRACTIONREQUEST_CFG

Name	P_TRACTIONREQUEST_CFG
Description	Traction request configuration

Name	P_TRACTIONREQUEST_CFG		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No Traction request available	
	1	Traction request implemented through FVA	
	2	Traction request implemented through TCMS	
Default value	0 No Traction request available		

5.2.61 P_TRAIN_DATA

Name	P_TRAIN_DATA		
Description	Train data configuration		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No train data available	
	1	Train data statically stored in FVA	
	2	Train data received from TCMS	
Default value	0 No train data available		

5.2.62 P_TRAIN_MASS

Name	P_TRAIN_MASS		
Description	Train mass Static data set		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 14 bit	0 t 0	15000 t 15000	1 t
Default value	0		

5.2.63 P_TRAIN_MASS_CFG

Name	P_TRAIN_MASS_CFG		
Description	Train mass parameters configuration		
Type	Minimum Value	Maximum Value	Resolution/ Formula
Unsigned Integer 3 bit			
Special/ Reserved Values	0	No train mass parameters available	
	1	Train mass parameters implemented dynamically through FVA	
	2	Train mass parameters implemented dynamically through TCMS	
	3	Train mass parameters implemented statically through FVA	
	4-7	Spare	
Default value	0 No train mass parameters available		

5.2.64 P_UTC_TIME_MASTER

Name	P_UTC_TIME_MASTER
Description	Configuration of master time

Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	UTM_TCMS_only	Only TCMS Time available
	1	UTM_TCMS_master	TCMS and ATO time available, TCMS is master
	2	UTM_ACPU_master	TCMS and ATO time available, ATO is master
Default value	0 Only TCMS Time available		

5.2.64.1 Note: See 5.5.1, Reference Time for time management.

5.2.65 P_WHEEL_DIAMETER_CFG

Name	**P_WHEEL_DIAMETER_CFG**		
Description	Wheel Diameter information configuration		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 2 bit			
Special/ Reserved Values	0	No wheel Diameter information available	
	1	Wheel Diameter information available via FVA	
	2	Wheel Diameter information available from TCMS	
Default value	0 No wheel Diameter information available		

5.3 Conversion of packets and variables for FFFIS Extension of Subset-139

5.3.1 Timing aspects

5.3.1.1 The ATO Core interface shall be independent of the physical properties of the Subset-139 FFFIS, including low-level timing aspects

5.3.1.2 Note: Different transport implementations (MVB, Ethernet…) come with different constraints and approaches on timing.

5.3.1.3 Minimum timing requirements concerning the ATO Core interface may differ depending on:

- Computer cycle
- Implementation of regulation loops

5.3.1.4 In order to ensure a deterministic behavior of ATO algorithms, all packets shall be timestamped.

5.3.1.5 Note: If TCMS and ATO are not based on the same time base, the FVA shall provide a synchronization feature. In this case, the master clock shall be configurable.

5.3.2 Mapping of packets

OCORA Extension	***Subset-139***	***Direction***
Packet Number 0: ATO Status	ATO_TCMS_data	ATO-TCMS
Packet Number 1: Propulsion (Traction / Dynamic Brake) Control		

OCORA Extension	***Subset-139***	***Direction***
Packet Number 2: Pneumatic and special brake control		
Packet Number 3: Holding Brake control		
Packet Number 5: Door control		
Packet Number 9: Config Info Request		
Packet 41: Direct Traction / Brake Commands*		
Packet Number 21: Propulsion (Traction / Dynamic Brake) Status	TCMS_ATO_data fast	TCMS-ATO
Packet Number 22: Pneumatic and special brake Status		
Packet Number 25: Odometry Data		
Packet Number 26: Door status		
Packet Number 23: Holding Brake status		
Packet Number 28: Train and vehicle status		
Packet Number 29: UTC Master Time		
Packet Number 27: Train and vehicle specific values	TCMS_ATO_data slow	TCMS-ATO
Packet Number 24: Brake Model*		
Packet Number 10: ATO Time*	none	ATO-FVA
Packet Number 31: TCMS Capabilities	none	FVA-ATO
Packet Number 32: Error Status		

*Optional packet

5.4 Functional concept of the ATO Functional Vehicle Adaptor

5.4.1 General

5.4.1.1 The FVA serves as an abstraction layer between the ATO Core Interface and the Subset-139 FFFIS

5.4.1.2 The FVA shall normalise timing information of the variables exchanged between the ATO and the TCMS

5.4.1.3 For packets sent from the ATO to the TCMS, the timing shall be determined by the physical time as seen by the FFFIS.

5.4.1.4 For packets received by the ATO from the TCMS that carry a TCMS timestamp, the FVA shall set the timestamp in the TCMS- ATO packets of this specification to the master UTC time reference (as described in chapter 6.5.1). If necessary, the TCMS timestamp shall be adjusted to the master UTC time reference

5.4.1.5 Note: The objective is to ensure that the ATO knows the exact time the respective variable was sent, in ATO time reference.

5.4.1.6 For packets received by the ATO from the TCMS that do not have a TCMS timestamp, the FVA shall set the timestamp in the TCMS- ATO packets of this specification to the master UTC time reference at the time of reception of the packet containing the related variable.

5.4.2 Interfaces and Data

5.4.2.1 The ATO Core Interface provides a packet- and variable- based language for the ATO. Based on this language, the ATO is able to implement all functionality as required by Subset-125 without any implicit knowledge about the specifics of the underlying TCMS. Specific packets inform the ATO about TCMS capabilities and configuration inconsistencies.

5.4.2.2 The Subset-139 FFFIS provides access to the TCMS.

5.4.2.3 The external function API defines data to be exchanged with optional external functions. These functions may either be implemented as software components or as external programmable electronic systems. The details of the implemention of these functions are project- specific.

5.4.3 Functional Dataflow

5.4.3.1 The FVA processes data using the following principles:

- Data is received from the source (ATO or TCMS)
- The FVA checks if a matching parameter is to be evaluated and if yes, processes the parameter
- Depending on the parameter,
    - The variable is selected to be sent directly to the sink (TCMS or ATO)
    - Or a local parameter is used to set the value of the variable before sending it to the sink (TCMS or ATO)
    - Or an external function is called / data sent to an external subsystem in order to calculate the value of the variable before sending it to the sink (TCMS or ATO)
- If required, the value is converted to match the format required by the sink
- The variable is sent to the sink (TCMS or ATO)

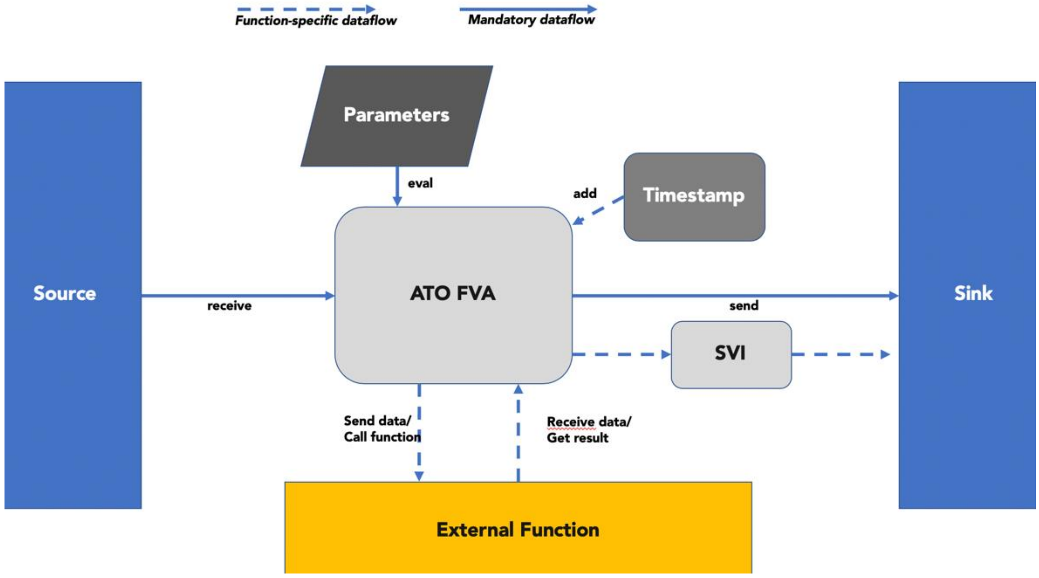


Figure 5: Principles of ATO FVA data flow

5.4.3.2 Explanation

The data flow in both directions. *Figure 5: Principles of ATO FVA data flow* is assuming that the source is the Unified and standardized interface for ATO OBU (OCORA 40-010) while the sink consinsts of the the SUBSETs -139 & -143
For the data flowing from the TCMS to the ATO, the same principles apply.

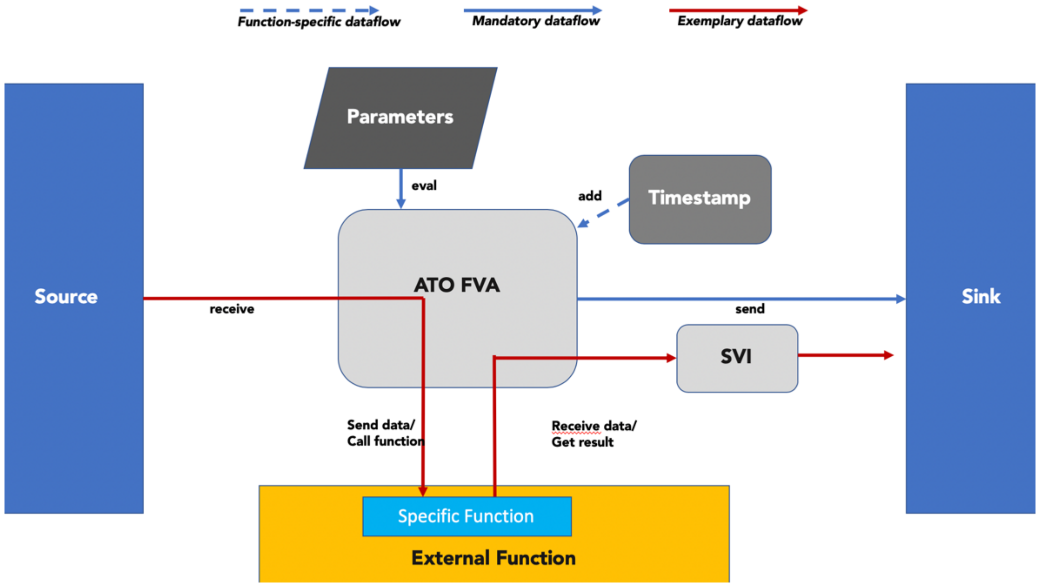


Figure 6: Example data flow through EXT and SVI

Data are sent from the source to the sink, through the FVA. Depending on the parameters, the data will either be:

- Not sent at all. (the data transfer is not possible with the specific configuration
- Sent directly to the Sink
- Sent to the Sink through the Specific Vehicle Interface (either the data are optional, or the concrete Subset-139 implementation does not support the data)
- Sent to the External Function interface for processing before the results will be forwarded to the Sink (the TCMS lacks the required functionality and/ or interface support)

Figure 6: Example data flow through EXT *and SVI* shows a situation where a specific EXT function needed to be added. In addition, the data flow through the SVI.
The parameters are set accordingly.

The EXT interface provides a standardized way to communicate with external functions.

5.5 List of ATO FVA Functions

5.5.1 Reference Time

5.5.1.1 Formal definition:

FVA::Time::Reference_Time

5.5.1.2 The FVA shall maintain a master time reference.

5.5.1.3 Note: some TCMS systems may provide a time reference, and/ timestamped odometry data. However, the ATO may have access to a more stable/ accurate time standard, for example through FRMCS or GNSS. In order to manage such configurations, we intoduce the notion of a reference time.

5.5.1.4 Note: Multiple control loops may exist in a traction and braking control system. Some possible loop configurations could include:

- Local loop in TCMS. The control algorithm has no external dependencies (for example: anti-skid system
- Loop across the system boundaries: the relative braking / traction command of the ATO interacts with the TCMS – based AFB regulation of the vehicle

It is important to base such algorthithms on a common time reference in order to avoid jitter and other artifacts arising from the noise that would disturb the control loops if we would couple control cycles and communication cycles.

5.5.1.5 All packets (and consequentially the variables contained in these packets) on the ATO Core interface carry a timestamp with a resolution of 1ms and a precision of greater than 0,01ms

5.5.1.6 TCMS Interface for Reference Time function

From TCMS	*To TCMS*	*Remark*
T_TCMS_UTC		
T_TCMS_UTC_MS		

5.5.1.7 ATO Interface for Reference Time function

From ATO	*To ATO*	*Remark*
PACKET 10: ATO TIME	**PACKET NUMBER 29: UTC MASTER TIME**	
UTC_TIME_ACPU	UTC_TIME_TCMS	
UTC_TIME_MS_ACPU	UTC_TIME_MS_TCMS	
	UTC_MASTER_TCMS	
	TIME_OFFSET_SIGN_TCMS	
	TIME_OFFSET_TCMS	
	TIME_OFFSET_MS_TCMS	

5.5.1.8 Relevant parameters for Reference Time function

Parameter name	*Remark*
P_UTC_TIME_MASTER	

5.5.1.9 Related external function interface

- None

5.5.1.10 Formal function description

5.5.1.10.1 Formal description: see Model

5.5.1.11 Functional description

5.5.1.11.1 If P_UTC_TIME_MASTER is set to `1: TCMS and ATO time available, TCMS is master`, the following rules apply:

Condition	*Value of output*	*Output*	*Remark*
	T_TCMS_UTC	UTC_TIME_TCMS	
	T_TCMS_UTC_MS	UTC_TIME_MS_TCMS	
	1: TCMS AND ATO TIME AVAILABLE, TCMS IS MASTER	UTC_MASTER_TCMS	
(T_TCMS_UTC + T_TCMS_UTC_MS) > (UTC_TIME_ACPU + UTC_TIME_MS_ACPU)	3: ATO TIME < TCMS TIME	TIME_OFFSET_SIGN_TCMS	
	(T_TCMS_UTC + T_TCMS_UTC_MS) - (UTC_TIME_ACPU + UTC_TIME_MS_ACPU)	TIME_OFFSET_TCMS, TIME_OFFSET_MS_TCMS	CALC_TIME_OFFSET: COMPLEX SUBSTRACTION
(T_TCMS_UTC + T_TCMS_UTC_MS) < (UTC_TIME_ACPU + UTC_TIME_MS_ACPU)	2: ATO TIME > TCMS TIME	TIME_OFFSET_SIGN_TCMS	
	(UTC_TIME_ACPU + UTC_TIME_MS_ACPU) - (T_TCMS_UTC + T_TCMS_UTC_MS)	TIME_OFFSET_TCMS TIME_OFFSET_MS_TCMS TIME_OFFSET_SIGN_TCMS	CALC_TIME_OFFSET: COMPLEX SUBSTRACTION
(T_TCMS_UTC + T_TCMS_UTC_MS) = (UTC_TIME_ACPU + UTC_TIME_MS_ACPU)	1 NO OFFSET	TIME_OFFSET_SIGN_TCMS	
	0	TIME_OFFSET_TCMS TIME_OFFSET_MS_TCMS	

5.5.1.12 If P_UTC_TIME_MASTER is set to 2: TCMS and ATO time available, ATO is master, the following rules apply:

Condition	*Value of output*	*Output*	*Remark*
	UTC_TIME_ACPU	UTC_TIME_TCMS	
	UTC_TIME_MS_ACPU	UTC_TIME_MS_TCMS	
	1: TCMS AND ATO TIME AVAILABLE, TCMS IS MASTER	UTC_MASTER_TCMS	
(T_TCMS_UTC + T_TCMS_UTC_MS) > (UTC_TIME_ACPU + UTC_TIME_MS_ACPU)	3: ATO TIME < TCMS TIME	TIME_OFFSET_SIGN_TCMS	
	(T_TCMS_UTC + T_TCMS_UTC_MS) - (UTC_TIME_ACPU + UTC_TIME_MS_ACPU)	TIME_OFFSET_TCMS, TIME_OFFSET_MS_TCMS	CALC_TIME_OFFSET: COMPLEX SUBSTRACTION
(T_TCMS_UTC + T_TCMS_UTC_MS) < (UTC_TIME_ACPU + UTC_TIME_MS_ACPU)	2: ATO TIME > TCMS TIME	TIME_OFFSET_SIGN_TCMS	
	(UTC_TIME_ACPU + UTC_TIME_MS_ACPU) - (T_TCMS_UTC + T_TCMS_UTC_MS)	TIME_OFFSET_TCMS TIME_OFFSET_MS_TCMS TIME_OFFSET_SIGN_TCMS	CALC_TIME_OFFSET: COMPLEX SUBSTRACTION
(T_TCMS_UTC + T_TCMS_UTC_MS) = (UTC_TIME_ACPU + UTC_TIME_MS_ACPU)	1 NO OFFSET	TIME_OFFSET_SIGN_TCMS	
	0	TIME_OFFSET_TCMS TIME_OFFSET_MS_TCMS	

5.5.1.13 If P_UTC_TIME_MASTER is set to 0: Only TCMS Time available, the following rules apply:

Condition	*Value of output*	*Output*	*Remark*
	T_TCMS_UTC	UTC_TIME_TCMS	
	T_TCMS_UTC_MS	UTC_TIME_MS_TCMS	
	0: ONLY TCMS TIME AVAILABLE	UTC_MASTER_TCMS	
	1 NO OFFSET	TIME_OFFSET_SIGN_TCMS	
	0	TIME_OFFSET_TCMS, TIME_OFFSET_MS_TCMS	

5.5.1.14 Variable mappings

In	*Out*	*Remark*
T_TCMS_UTC	UTC_TIME_TCMS	uint32 value propagated without any conversion
UTC_TIME_ACPU		
UTC_TIME_MS_ACPU	UTC_TIME_MS_TCMS	uint32 value propagated without any conversion
T_TCMS_UTC_MS		
	UTC_MASTER_TCMS	Variables generated by function
	TIME_OFFSET_SIGN_TCMS	
	TIME_OFFSET_TCMS	

5.5.2 ATO_STATE_ACPU

5.5.2.1 Formal definition:

5.5.2.1.1 See model

5.5.2.2 FVA::ATO_Status::ATOState

5.5.2.3 The ATO State Message shall be forwarded to the TCMS

5.5.2.4 ATO Interface for ATO State function

From ATO	*To ATO*	*Remark*
PACKET 0: ATO STATUS		
ATO_STATE_ACPU		

5.5.2.5 TCMS Interface for ATO State function

From TCMS	*To TCMS*	*Remark*
	M_ATO_STATE	
	M_ATO_CONFIG	

5.5.2.6 Relevant parameters for ATO State function

Parameter name	*Remark*
	No parameters

5.5.2.7 Related external function interface

- None

5.5.2.8 Formal function description

5.5.2.8.1 See model

5.5.2.9 Functional description

5.5.2.9.1 ATO_State is continually being forwarded to the TCMS cyclically

5.5.2.10 Variable mappings

In	*Out*	*Remark*
ATO_STATE_ACPU	M_ATO_STATE	
ATO_STATE_NP	0B00000001	
ATO_STATE_CO	0B00000010	
ATO_STATE_NA	0B00000100	
ATO_STATE_AV	0B00001000	
ATO_STATE_RE	0B00010000	
ATO_STATE_EG	0B00100000	
ATO_STATE_DE	0B01000000	
ATO_STATE_FA	0B10000000	
	0B00000000	DEFAULT

5.5.3 ATO_CONFIG_ACPU

5.5.3.1 Formal definition:

FVA::ATOStatus::ATOConfig

5.5.3.2 The ATO Config Message shall be forwarded to the TCMS, if the TCMS is able to process it. Otherwise, an error shall be raised and sent back to the ATO.

5.5.3.2.1 Note: If the ATO is unable to process error messages, this feature may be switched off globally by setting the parameter P_ERRORS to 0.

5.5.3.3 ATO Interface for ATO Config function

From ATO	***To ATO***	***Remark***
PACKET 0: ATO STATUS	**PACKET 32: ERROR STATUS**	
ATO_CONFIG_ACPU	ERROR_TRACTION_OPTION_1_NOT_CFG	
	ERROR_TRACTION_OPTION_2_NOT_CFG	

5.5.3.4 TCMS Interface for ATO Config function

From TCMS	***To TCMS***	***Remark***
	M_ATO_Config	

5.5.3.5 Relevant parameters for ATO Config function

Parameter name	***Remark***
P_Traction_Option_1_cfg	
P_Traction_Option_2_cfg	

5.5.3.6 Related external function interface

- None

5.5.3.7 Formal function description

5.5.3.7.1 See model

5.5.3.8 Functional description

Condition	***Value of output***	***Output***	***Remark***
ATO_CONFIG_ACPU = ATO_CONFIG_Option1 and P_Traction_Option_1_cfg = true	False	ERROR_TRACTION_OPTION_1_NOT_CFG	
	c_BITSET8_0	M_ATO_CONFIG	
ATO_CONFIG_ACPU = ATO_CONFIG_Option1 and P_Traction_Option_1_cfg = false	True	ERROR_TRACTION_OPTION_1_NOT_CFG	
	c_BITSET8_default	M_ATO_CONFIG	
ATO_CONFIG_ACPU = ATO_CONFIG_Option2 and P_Traction_Option_2_cfg = true	False	ERROR_TRACTION_OPTION_2_NOT_CFG	
	c_BITSET8_0	M_ATO_CONFIG	
ATO_CONFIG_ACPU = ATO_CONFIG_Option2 and P_Traction_Option_2_cfg = false	True	ERROR_TRACTION_OPTION_2_NOT_CFG	
	c_BITSET8_default	M_ATO_CONFIG	

5.5.3.9 Variable mappings

In	*Out*	*Remark*
ATO_CONFIG_ACPU	M_ATO_Config	
ATO_CONFIG_Option1	0	
ATO_CONFIG_Option2	1	
ATO_CONFIG_no_option	2	Also default mapping
	ERROR_TRACTION_OPTION_1_NOT_CFG	Variables generated by function
	ERROR_TRACTION_OPTION_2_NOT_CFG	

5.5.4 Relative Traction/ Brake Control

5.5.4.1 Formal definition:

5.5.4.2 FVA::Propulsion::RelativeTractionRequest

5.5.4.3 The Relative Traction/ Brake Request shall be forwarded to the TCMS, if the TCMS is able to process it. Otherwise, an error shall be raised and sent back to the ATO.

5.5.4.4 The auxiliary signals Traction Request and Brake Request shall be processed using the same principles as the Relative Traction/ Brake Request

5.5.4.5 It shall be possible to integrate or connect an external function in cases which the TCMS cannot process the data. This external function shall in this case be able to control traction and brake directly using binary signals.

5.5.4.5.1 Note: If the ATO is unable to process error messages, this feature may be switched off globally by setting the parameter P_ERRORS to 0.

5.5.4.6 ATO Interface for Relative Traction/ Brake Request function

From ATO	*To ATO*	*Remark*
PACKET 1: PROPULSION		
RELATIVE_TRACTION_REQUEST_ACPU		
TractionRequest_ACPU		
BrakeRequest_ACPU		
	ERR_RelTractionRequest_not_cfg	

5.5.4.7 TCMS Interface for Relative Traction/ Brake Request function

From TCMS	*To TCMS*	*Remark*
	M_ATO_RTBRq	
	Q_ATO_AuxTB	
	AD_BINARY_RELEASE_INDIRECT_BRAKE_ACPU	*
	AD_BINARY_ENGAGE_INDIRECT_BRAKE_ACPU	*
	AD_BINARY_TRACTION_UP_ACPU	*
	AD_BINARY_TRACTION_DOWN_ACPU	*
	AD_BINARY_TRACTION_0_ACPU	*
	AD_BINARY_RELEASE_DIRECT_BRAKE_ACPU	*
	AD_BINARY_ENGAGE_DIRECT_BRAKE_ACPU	*
	TractionRequest_ACPU_AD	*
	BrakeRequest_ACPU_AD	*
	DYNAMICBRAKEREQUEST_X_ACPU	*

5.5.4.8 Relevant parameters for Relative Traction/ Brake Request function

Parameter	*Remark*
P_RelTractionRequest_cfg	

5.5.4.9 Related external function interface for Relative Traction/ Brake Request

Function Name	*Variables sent to EXT function*	*Variables received back*
EXT_Control_RelTractionRequest	ACPU_EXT_Relative_Traction_Req	EXT_Release_Indirect_Brake*
	RTR_X_ACPU	EXT_Enage_Indirect_Brake*
		TRACTION_UP_BIN_X_EXT*
		TRACTION_DOWN_BIN_X_EXT*
		TRACTION_0_BIN_X_EXT*
		EXT_Release_Direct_Brake*
		EXT_Engage_Direct_Brake*
		DYNAMICBRAKEREQUEST_X_EXT*

* Optional, if external function is used

5.5.4.10 Formal function description

5.5.4.10.1 See model

5.5.4.11 Functional description

Condition	*Value of output*	*Output*	*Remark*
P_RELTRACTIONREQUEST _CFG = PNVT_FVA	true	RTR_X_ACPU	*
	TractionRequest_ACPU	TractionRequest_ACPU_AD	*
	BrakeRequest_ACPU	BrakeRequest_ACPU_AD	*
	RELATIVE_TRACTION_REQUEST _ACPU	ACPU_EXT_Relative_Traction_Req	*
	EXT_Release_Indirect_Brake	AD_BINARY_RELEASE_INDIRECT_ BRAKE_ACPU	*
	EXT_Engage_Indirect_Brake	AD_BINARY_ENGAGE_INDIRECT_ BRAKE_ACPU	*
	TRACTION_UP_BIN_X_EXT	AD_BINARY_TRACTION_UP_ ACPU	*
	TRACTION_DOWN_BIN_X_EXT	AD_BINARY_TRACTION_DOWN_ ACPU	*
	TRACTION_0_BIN_X_EXT	AD_BINARY_TRACTION_0_ ACPU	*
	EXT_Release_Direct_Brake	AD_BINARY_RELEASE_DIRECT_ BRAKE_ACPU	*
	EXT_Engage_Direct_Brake	AD_BINARY_ENGAGE_DIRECT_ BRAKE_ACPU	*
	DYNAMICBRAKEREQUEST_X_ EXT	DYNAMICBRAKEREQUEST_X_ ACPU	*
P_RELTRACTIONREQUEST _CFG = PNVT_TCMS	RELATIVE_TRACTION_REQUEST _ACPU	M_ATO_RTBRq	
P_RELTRACTIONREQUEST _CFG = PNVT_TCMS and TractionRequest_ACPU = true	c_BITSET8_0	L_Q_ATO_AuxTB_1	
P_RELTRACTIONREQUEST _CFG = PNVT_TCMS and TractionRequest_ACPU = false	c_BITSET8_default	L_Q_ATO_AuxTB_1	
P_RELTRACTIONREQUEST _CFG = PNVT_TCMS and BrakeRequest_ACPU= true	c_BITSET8_1	L_Q_ATO_AuxTB_2	
P_RELTRACTIONREQUEST _CFG = PNVT_TCMS and BrakeRequest_ACPU= false	c_BITSET8_default	L_Q_ATO_AuxTB_2	
	0	M_ATO_RTBRq	

Condition	*Value of output*	*Output*	*Remark*
P_RELTRACTIONREQUEST_CFG = PNVT_None	c_BITSET8_default	Q_ATO_AuxTB	
	true	ERR_RelTractionRequest_not_cfg	
P_RELTRACTIONREQUEST_CFG = PNVT_FVA or P_RELTRACTIONREQUEST_CFG = PNVT_TCMS	L_Q_ATO_AuxTB_1 or L_Q_ATO_AuxTB_2	Q_ATO_AuxTB	Bitwise OR

5.5.4.11.1 Any values not explicitly set in above table shall be set to their respective default values (see formal definition)

5.5.5 Traction / Brake Control Status

5.5.5.1 Formal definition:

5.5.5.2 FVA::Propulsion::RelativeTractionStatus

5.5.5.3 The Relative Traction/ Brake Status shall be forwarded to the ATO, if the TCMS is able to generate them.

5.5.5.4 It shall be possible to integrate or connect an external function in cases the TCMS cannot produce the data.

5.5.5.5 TCMS Interface for Relative Traction/ Brake Request function

From TCMS	*To TCMS*	*Remark*
Q_TCMS_AuxTB		

5.5.5.6 ATO Interface for Relative Traction/ Brake Request function

From ATO	*To ATO*	*Remark*
	Traction_Ready_TCMS	
	ENGAGEMENT_READY_TCMS	
	TractionApplied	

5.5.5.7 Relevant parameters for Relative Traction/ Brake Request function

Parameter	*Remark*
P_RelTractionRequest_cfg	

5.5.5.8 Related external function interface for Relative Traction/ Brake Request

Function Name	*Variables sent to EXT function*	*Variables received back*
EXT_Control_RelTractionStatus*	RTS_X_ACPU*	Traction_Ready_EXT*
		ENGAGEMENT_READY_EXT*
		TractionApplied_EXT*

* Optionally, if external function is used

5.5.5.9 Formal function description

5.5.5.1 Functional description

Condition	*Value of output*	*Output*	*Remark*
P_RELTRACTIONREQUEST_CFG = PNVT_FVA	true	RTS_X_ACPU	
	Traction_Ready_EXT	Traction_Ready_TCMS	

Condition	Value of output	Output	Remark
	ENGAGEMENT_READY_EXT	ENGAGEMENT_READY_TCMS	
	TractionApplied_EXT	Traction_Applied_TCMS	
P_RELTRACTIONREQUEST _CFG = PNVT_TCMS	(Q_TCMS_AuxTB and c_BITSET8_0) <> 0	Traction_Ready_TCMS	Bitwise and
	(Q_TCMS_AuxTB and c_BITSET8_2) <> 0	ENGAGEMENT_READY_TCMS	Bitwise and
	(Q_TCMS_AuxTB and c_BITSET8_3) <> 0	Traction_Applied_TCMS	Bitwise and

5.5.5.1.1 Any values not explicitly set in above table shall be set to their respective default values (see formal definition)

5.5.6 Dynamic Brake Status

5.5.6.1 FVA::Propulsion::DynamicBrakeStatus

5.5.6.2 The Dynamic Brake Status shall be forwarded to the ATO, if the TCMS is able to generate it.

5.5.6.3 It shall be possible to integrate or connect an external function in cases the TCMS cannot produce the data.

5.5.6.4 TCMS Interface for Relative Traction/ Brake Request function

From TCMS	To TCMS	Remark
Q_TCMS_AuxTB		

5.5.6.5 ATO Interface for Relative Traction/ Brake Request function

From ATO	To ATO	Remark
	DYNAMIC_BRAKE_APPLIED_TCMS	
	DYNAMIC_BRAKE_AVAILABLE_TCMS	
	DYNAMIC_BRAKE_READY_TCMS	

5.5.6.6 Relevant parameters for Relative Traction/ Brake Request function

Parameter	Remark
P_DynamicBrake_cfg	

5.5.6.7 Related external function interface for Relative Traction/ Brake Request

Function Name	Variables sent to EXT function	Variables received back
EXT_DynamicBrakeStatus	DBS_X_ACPU	DYNAMIC_BRAKE_APPLIED_X_EXT
		DYNAMIC_BRAKE_READY_X_EXT

* Optional, if external function is used

5.5.6.8 Formal function description

5.5.6.9 Functional description

Condition	Value of output	Output	Remark
P_DynamicBrake_cfg = PNVT_FVA	true	RTS_X_ACPU	
	DYNAMIC_BRAKE_APPLIED_X_EXT	DYNAMIC_BRAKE_APPLIED_ TCMS	
	true	DYNAMIC_BRAKE_AVAILABLE_ TCMS	
	DYNAMIC_BRAKE_READY_X_EXT	DYNAMIC_BRAKE_READY_	

Condition	*Value of output*	*Output*	*Remark*
		TCMS	
P_DynamicBrake_cfg = PNVT_TCMS	(Q_TCMS_AuxTB and c_BITSET8_4) <> 0	DYNAMIC_BRAKE_APPLIED_TCMS	Bitwise and
	true	DYNAMIC_BRAKE_AVAILABLE_TCMS	
	(Q_TCMS_AuxTB and c_BITSET8_1) <> 0	DYNAMIC_BRAKE_READY_TCMS	Bitwise and

5.5.6.9.1 Any values not explicitly set in the above table shall be set to their respective default values (see formal definition).

5.5.7 Pneumatic Brake Control

5.5.7.1 The Pneumatic Brake Control commands shall be forwarded to the TCMS, if the TCMS is able to process them. Otherwise, an error shall be raised and sent back to the ATO.

5.5.7.2 It shall be possible to integrate or connect an external function in cases the TCMS cannot process the data. This external function shall in this case be able to the pneumatic brakes directly using binary signals.

5.5.7.2.1 Note: If the ATO is unable to process error messages, this feature may be switched off globally by setting the parameter P_ERRORS to 0.

5.5.7.3 ATO Interface for Pneumatic Brake Control function

From ATO	*To ATO*	*Remark*
PACKET 2: PNEUMATIC		
INDIRECT_BRAKE_REQUEST_ACPU		
DIRECT_BRAKE_REQUEST_ACPU		
RELEASE_QUICK_BRAKE_ACPU		
	PNEUBRAKE_NOT_CFG_Error	

5.5.7.4 TCMS Interface for Pneumatic Brake Control function

From TCMS	*To TCMS*	*Remark*
	M_ATO_IndiBRq	
	M_ATO_DirBRq	
	AD_BINARY_RELEASE_DIRECT_BRAKE_ACPU	*
	AD_BINARY_ENGAGE_DIRECT_BRAKE_ACPU	*
	AD_BINARY_RELEASE_INDIRECT_BRAKE_ACPU	*
	AD_BINARY_ENGAGE_INDIRECT_BRAKE_ACPU	*
	AD_BINARY_LOW_PRESSURE_OVERFILLING_ACPU	*
	AD_ACPU_HIGH_PRESSURE_FILLING_ACPU	*

5.5.7.5 Relevant parameters for Relative Traction/ Brake Request function

Parameter	*Remark*
P_PNEUBRAKE_cfg	

5.5.7.6 Related external function interface for Relative Traction/ Brake Request

Function Name	*Variables sent to EXT function*	*Variables received back*
EXT_Control_Pneubrake Request	PBR_X_ACPU	EXT_Release_Indirect_Brake*
	INDIRECT_BRAKE_REQUEST_X_ACPU	EXT_Enage_Indirect_Brake*
	DIRECT_BRAKE_REQUEST_X_ACPU	EXT_Release_Direct_Brake*
	QUICK_BRAKE_RELEASE_X_ACPU	EXT_Engage_Direct_Brake*
		LOW_PRESSURE_OVERFILLING_X_EXT*
		HIGH_PRESSURE_FILLING_X_EXT*

* Optional, if external function is used

5.6 Pneumatic and special brake control

5.6.1 Immediate indirect air brake request

5.6.1.1 Overview

API Message	***From- To***	***Sequence/ Activation***	***Rules***
Indirect_Brake_Request	ATO to TCMS	Cyclic	Specific logic, see below

5.6.1.2 If the TCMS is unable to process a percentage- based Indirect Brake Request Command, the ATO can instead directly control the corresponding valve on the vehicle. In this case the signal `Indirect_Brake_Request_binary` shall be used.

5.6.1.3 Rules / Sequence

API Message	***From- To***	***Activation***
INDIRECT_BRAKE_REQUEST	**ATO TO FVA**	**RECEIVED FROM ATO**
ERROR_RELINDIRECTBRAKE_NOT_CFG	FVA to ATO	Sent if P_Rel_IndirectBrake_cfg is set to No and Indirect_Brake_Request is received from the ATO
[17] 6.2.4.8 Immediate indirect air brake request	FVA to TCMS	Sent if P_Rel_IndirectBrake_cfg is set to TCMS
Function Control_Binary_Indirect_Brake_Request	n/a	Called if P_Rel_IndirectBrake_cfg is set to FVA

5.6.2 Indirect Binary Brake Request (digital)

5.6.2.1 It shall be possible to send a signal (FVA to TCMS) that directly controls the function “Release Indirect Brake”.

5.6.2.2 The additional TCMS signal “`Release_Indirect_Brake_Bin`”

5.6.2.3 A project- specific or parameterizable function “Control_Binary_Indirect_Brake_Request” shall be provided by the FVA in this case.

Function	***Interface***	***Description***
CONTROL_BINARY_INDIRECT_BRAKE_REQUEST		
	Indirect_Brake_Request	to Input from ATO
	Brake_Pipe_Pressure	to Input from TCMS
	Release_Indirect_Brake_Bin	Output to TCMS (Additional variable)
Functional Description	Project- specific regulator function controlling the variable in order to achieve alignment of the variable Brake_Pipe_Pressure with the variable Indirect_Brake_Request	

Table 1: Indirect Binary Brake Request (digital) functional vehicle adaptor interface

5.6.2.4 For trains with single- release braking systems, project- specific, appropriate control logic must be foreseen in order to ensure that the appropriate brake configuration and braking pressures are ensured.

5.6.3 Immediate direct air brake request

5.6.3.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`Direct_Brake_Request`	ATO to TCMS	Sporadic: On change of value	Specific logic, see below

Table 2: Direct air brake request functional vehicle adaptor interface

5.6.3.2 If the TCMS is unable to process a percentage- based Indirect Brake Request Command, the ATO can instead directly control the corresponding valve on the vehicle. In this case the signal `Indirect_Brake_Request_binary` shall be used.

5.6.3.3 Rules / Sequence

API Message	From- To	Activation
`Indirect_Brake_Request`	ATO to FVA	Received from ATO
`ERROR_RELINDIRECTBRAKE_NOT_CFG`	FVA to ATO	Sent if `P_Rel_IndirectBrake_cfg` is set to `No` and `Indirect_Brake_Request` is received from the ATO
[17] 6.2.4.8 Immediate indirect air brake request	FVA to TCMS	Sent if `P_Rel_IndirectBrake_cfg` is set to `TCMS`
Function `Control_Binary_Indirect_Brake_Request`	n/a	Called if `P_Rel_IndirectBrake_cfg` is set to `FVA`

Table 3: Direct air brake request logic

5.6.4 Direct Binary Brake Request (digital)

5.6.4.1 It shall be possible to send a signal (FVA to TCMS) that directly controls the function “Release Indirect Brake”.

5.6.4.2 The additional TCMS signal “`Release_Indirect_Brake_Bin`”

5.6.4.3 A project- specific or parameterizable function “Control_Binary_Indirect_Brake_Request” shall be provided by the FVA in this case.

Function	Interface	Description

Control_Binary_Indirect_Brake_Request		
	Direct_Brake_Request	Input from ATO
	Brake_Pipe_Pressure	Input from TCMS
	Engage_Indirect_Brake_Bin	Output to TCMS (Additional variable)
	Release_Indirect_Brake_Bin	Output to TCMS (Additional variable)
Functional Description	Project- specific regulator function controlling the variable in order to achieve alignment of the variable Brake_Pipe_Pressure with the variable Indirect_Brake_Request	

Table 4: Direct Binary Brake Request (digital) function

5.6.4.4 For trains with single- release braking systems, project- specific and appropriate control logic must be foreseen in order to ensure that the appropriate brake configuration and braking pressures are ensured.

5.6.5 Quick brake release request

5.6.5.1 Overview

API Message	From- To
Release_Quick_Brake	ATO to TCMS

Table 5: Quick Brake Release Request functional vehicle adaptor interface

5.6.5.2 Rules / Sequence

API Message	From- To	Activation
Release_Quick_Brake	ATO to FVA	Received from ATO
ERROR_RELQUICKBRAKE_NOT_CFG	FVA to ATO	Sent if P_Rel_QuickBrake_cfg is set to No and Indirect_Brake_Request is received from the ATO
[17] 6.2.4.8 Quick brake release request	FVA to TCMS	Passed through from ATO ifP_Rel_QuickBrake_cfg is set to TCMS
Function Release_QuickBrake_LL	FVA to TCMS	Called if P_Rel_QuickBrake_cfg is set to FVA

Table 6: Quick Brake logic

5.6.6 Quick Brake Release Request (low- level)

5.6.6.1 It shall be possible to send a signal (FVA to TCMS) that directly controls the function “Release Quick Brake” through low- level functions.

5.6.6.2 The additional TCMS signal “`Low_pressure_overfilling`” shall be used by this function

5.6.6.3 The additional TCMS signal “`High_pressure_filling`” shall be used by this function

5.6.6.4 A project- specific or parameterizable function “Quick Brake Release Request_LL” shall be provided by the FVA in this case.

Function	Interface	Description
`Quick Brake Release Request_LL`		
	`Release_Quick_Brake`	Input from ATO
	`Low_pressure_overfilling`	Output to TCMS (Additional variable)
	`High_pressure_filling`	Output to TCMS (Additional variable)
Functional Description	Project- specific function controlling for implementing a Quick Brake Release Request functionality	

Table 7: Quick Brake Relase Request Low- Level Function

5.6.1 EB released

5.6.1.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`EB_Released`	TCMS to ATO	Sporadic, Triggered by status change in the TCMS	Passed through

Table 8: EB Released functional vehicle adaptor interface

5.6.2 SB applied

5.6.2.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`SB_Applied`	TCMS to ATO	Sporadic, Triggered by status change in the TCMS	Passed through

Table 9: SB Applied functional vehicle adaptor interface

5.6.3 Traction over brake enabled

5.6.3.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`Traction_Over_Brake`	TCMS to ATO	Sporadic, Triggered by status change in the TCMS	Specific logic, see below

Table 10: Traction Over Brake Enabled functional vehicle adaptor interface

5.6.3.2 Rules / Sequence

API Message	From- To	Activation
[17] 6.2.3.1 Traction over brake enabled	TCMS to FVA	Received from TCMS
`Traction_Over_Brake`	FVA to ATO	Permanently set to false if `P_Traction_Over_Brake_cfg` is set to `No`
`Dynamic_Brake_Applied quick`	FVA to ATO	Passed through from TCMS if `P_Traction_Over_Brake_cfg` is set to `TCMS`
Function `Control_LL_RelTractionRequest`	n/a	Called if `P_Traction_Over_Brake_cfg` is set to `FVA`

Table 11: Traction Over Brake Enabled logic

5.6.4 Brake pipe pressure

5.6.4.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`Brake_Pipe_Pressure`	TCMS toATO	Cyclic	Passed through

Table 12: Brake Pipe Pressure functional vehicle adaptor interface

5.6.5 Pressure at brake distributor output

5.6.5.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`Brake_Distributor_Pressure`	TCMS to ATO	Cyclic	Passed through

Table 13: Pressure at brake distributor functional vehicle adaptor interface

5.6.6 Direct brake applied

5.6.6.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Direct_Brake_Applied	TCMS to ATO	Cyclic	Passed through

Table 14: Direct brake applied functional vehicle adaptor interface

5.7 Holding Brake

5.7.1 Holding brake request

5.7.1.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Holding_Brake_Request	ATO to TCMS	Sporadic, Triggered by ATO	Specific logic, see below

Table 15: Holding brake request functional vehicle adaptor interface

5.7.1.2 Rules / Sequence

API Message	From- To	Activation
Holding_Brake_Request	ATO to FVA	Received from ATO
ERROR_HOLDINGBRAKE_NOT_CFG	FVA to ATO	Sent if P_Holding_Brake_cfg is set to No and Holding_Brake_Request is received from the ATO
[17] 6.2.4.12 Holding brake request	FVA to TCMS	Sent to TCMS if P_Holding_Brake_cfg is set to TCMS and Holding_Brake_Request is received from the ATO
Function Control_HoldingBrake_Request_LL	n/a	Called if P_Holding_Brake_cfg is set to FVA and Holding_Brake_Request is received

Table 16: Holding brake request logics

5.7.2 Holding brake applied

5.7.2.1 Overview

API Message	From- To	Sequence/	Rules

		Activation	
Holding_Brake_Applied	TCMS to ATO	Sporadic, Triggered by ATO	Specific logic, see below

Table 17: Holding brake applied functional vehicle adaptor interface

5.7.2.2 Rules / Sequence

API Message	From-To	Activation
[17] 6.2.4.12 Holding brake applied	TCMS to FVA	Received from TCMS
ERR_HoldingBrakeApplied_not_cfg	FVA to ATO	Sent if P_Holding_Brake_cfg is set to No and Holding_Brake_Applied is received from the TCMS
Holding_Brake_Applied	FVA to ATO	Sent to TCMS if P_Holding_Brake_cfg is set to TCMS and Holding_Brake_Applied is received from the TCMS
Function Control_HoldingBrake_Request_LL	n/a	Called if P_Holding_Brake_cfg is set to FVA and Holding_Brake_Applied is received from the TCMS

Table 18: Holding brake applied logic

5.7.3 Holding Brake Request (low- level)

5.7.3.1 It shall be possible to send a signal (FVA to TCMS) that directly controls the function “Holding Brake Request” through low- level functions in cases the TCMS does not directly support a holding brake implementation.

5.7.3.2 Appropriate low- level interfaces and functions may be used to implement (direct brake, indirect brake etc.)

5.7.3.3 Note: Not all variables referenced in Table 16: Holding brake request logics and Table 18: Holding brake applied logic need to be controlled by this function. The actual used signals and variables are project- specific.

5.7.3.4 A project- specific or parameterizable function “Control HoldingBrake Request LL” shall be provided by the FVA in this case.

Function	Interface	Description
Control_HoldingBrake_Request_LL		
	Holding_Brake_Request	Input from ATO
	[17] 6.2.4.12 Holding brake request	Output to TCMS (if appropriate)

Function	Interface	Description
	[17] 6.2.4.12 Holding brake applied	Input from TCMS (if available)
	`Holding_Brake_Request`	Output to ATO
	Additional project- specific functions and existing brake control variables (Project-Specific)	Input from TCMS (if appropriate) Output to TCMS (if appropriate)
Functional Description	Project- specific function controlling for implementing a Holding Brake Control functionality	

5.8 Odometry information

5.8.1 Actual speed

5.8.1.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`Actual_Speed`	TCMS to ATO	Cyclic	Passed through

Table 19: Actual speed functional vehicle adaptor interface

5.8.2 Actual acceleration

5.8.2.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`Actual_Acceleration`	TCMS to ATO	Cyclic	Passed through

Table 20: Actual acceleration functional vehicle adaptor interface

5.8.3 Travelled distance

5.8.3.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`Travelled_Distance`	TCMS to ATO	Cyclic	Passed through

Table 21: Travelled distance functional vehicle adaptor interface

5.8.4 TSI standstill

5.8.4.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`TSI_Standstill`	TCMS to ATO	Cyclic	Passed through

Table 22: TSI standstill functional vehicle adaptor interface

5.9 Door control signals

5.9.1 Door info request

5.9.1.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`Door_Info_Request`	ATO to FVA	Sporadic, Triggered by ATO	Specific logic, see below

Table 23: Door info request functional vehicle adaptor interface

5.9.1 Door info

5.9.1.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`Door_Info`	FVA to ATO	Sporadic	Specific logic, see below

Table 24: Door info functional vehicle adaptor interface

5.9.1.2 Rules / Sequence

API Message	From- To	Activation
`Door_Info_Request`	ATO to FVA	Received from ATO
`Door_Info`	FVA to ATO	Sent to ATO, triggered by reception of `Door_Info_Request`

Table 25: Door info request logic

5.9.2 Door enable request

5.9.2.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`DOOR_ENABLE_REQUEST_ACPU`	ATO to TCMS	Sporadic, Triggered by ATO	Specific logic, see below

Table 26: Door enable request functional vehicle adaptor interface

5.9.2.2 Rules / Sequence

API Message	From-To	Activation	Reference
DOOR_ENABLE_REQUEST_ACPU	ATO to FVA	Received from ATO	
ERROR_DOORENABLE_NOT_CFG	FVA to ATO	Sent if P_DOORENABLE_CFG is set to No and DOOR_ENABLE_REQUEST_ACPU is received from the ATO	
[17] 6.2.6.1 Table 6 Door enable request	FVA to TCMS	Sent to TCMS when DOOR_ENABLE_REQUEST_ACPU is received and P_DOORENABLE_CFG is set to TCMS	
Door_Enable_Request_LL	FVA to TCMS	Sent to TCMS when DOOR_ENABLE_REQUEST_ACPU is received and P_DOORENABLE_CFG is set to TCMS_advanced	

Table 27. Door enable request logic

5.9.2.3 If `P_DOORENABLE_CFG` is set to `TCMS` only the following information is passed to the TCMS:

- Doors enabled (left)
- Doors enable (right)
- Doors enable (outside)
- Doors enable (inside)

5.9.2.4 If `P_DOORENABLE_CFG` is set to `TCMS_advanced`, then the additional variable `Door_Enable_Request_LL` shall be used to control the permission for passengers to open individual doors.

5.9.3 Door open request

5.9.3.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Individual_Door_Open_Request_Left	ATO to TCMS	Sporadic, Triggered by ATO	Specific logic, see below
Individual_Door_Open_Request_Right	ATO to TCMS	Sporadic, Triggered by ATO	Specific logic, see below
Global_Door_Open_Request_Left	ATO to TCMS	Sporadic, Triggered by ATO	Specific logic, see below
Global_Door_Open_Request_Right	ATO to TCMS	Sporadic, Triggered by ATO	Specific logic, see below

Table 28: Door open request functional vehicle adaptor interface

5.9.3.2 Rules / Sequence

API Message	From-To	Activation
Individual_Door_Open_Request_Left	ATO to FVA	Received from ATO
Individual_Door_Open_Request_Right	ATO to	Received from ATO

	FVA	
Global_Door_Open_Request_Left	ATO to FVA	Received from ATO
Global_Door_Open_Request_Right	ATO to FVA	Received from ATO
ERROR_DOORCONTROL_NOT_CFG	FVA to ATO	Sent if P_Door_CONFIG is set to No and any …_Door_Open_Request_Left or …_Door_Open_Request_Right is received from the ATO
ERR_IndividualDoorControl_not_cfg	FVA to ATO	Sent if P_Door_CONFIG is set to TCMS and Individual_Door_Open_Request_Left or Individual_Door_Open_Request_Right is received from the ATO
[17] 6.2.6.1 Door open request	FVA to TCMS	Sent if P_Door_CONFIG is set to TCMS and Door_Open_Request_Left or Door_Open_Request_Right is received from the ATO
Individual_Door_Open_Request_Left	FVA to TCMS	Sent if P_Door_CONFIG is set to TCMS_advanced and Individual_Door_Open_Request_Left is received from the ATO
Individual_Door_Open_Request_Right	FVA to TCMS	Sent if P_Door_CONFIG is set to TCMS_advanced and Individual_Door_Open_Request_Right is received from the ATO

Table 29: Door open request logic

5.9.4 Door close request

5.9.4.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Door_Close_Request_Left	ATO to TCMS	Sporadic, Triggered by ATO	Specific logic, see below
Door_Close_Request_Right	ATO to TCMS	Sporadic, Triggered by ATO	Specific logic, see below

Table 30: Door close request functional vehicle adaptor interface

5.9.4.2 Rules / Sequence

API Message	From- To	Activation
Door_Close_Request_Left	ATO to FVA	Received from ATO
Door_Close_Request_Right	ATO to FVA	Received from ATO
ERROR_DOORCONTROL_NOT_CFG	FVA to ATO	Sent if P_Door_CONFIG is set to No and Door_Close_Request_Left or Door_Close_Request_Right is received from the ATO
[17] 6.2.6.1 Door close request	FVA to TCMS	Sent if P_Door_CONFIG is set to TCMS or TCMS_advanced and Door_Close_Request_Left or Door_Close_Request_Right is received from the ATO

Table 31: Door close request logic

5.9.5 Door status signals

5.9.5.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`Door_Status_Left`	TCMS to ATO	Sporadic, Triggered by TCMS	Specific logic, see below
`Door_Status_Right`	TCMS to ATO	Sporadic, Triggered by TCMS	Specific logic, see below

Table 32: Door Status functional vehicle adaptor interface

5.9.5.2 Rules / Sequence

API Message	From- To	Activation
`Door_Status_Left_LL`	TCMS to FVA	Received from TCMS (only relevant if `P_Door_CONFIG` is set to `TCMS_advanced`)
`Door_Status_Right_LL`	TCMS to FVA	Received from TCMS (only relevant if `P_Door_CONFIG` is set to `TCMS_advanced`)
[17] 6.2.6.1 Door status signals	TCMS to FVA	Received from TCMS (only relevant if `P_Door_CONFIG` is set to `TCMS`)
`Door_Status_Left`	FVA to ATO	Sent to ATO when related door status data are received from the TCMS
`Door_Status_Right`	FVA to ATO	Sent to ATO when related door status data are received from the TCMS

Table 33: Door status signal logic

5.9.5.3 If `P_Door_CONFIG` is set to `TCMS_advanced` then the FVA shall forward the information received via the variables `Door_Status_Left_LL` and `Door_Status_Right_LL`.

5.9.5.4 If `P_Door_CONFIG` is set to `TCMS` then the FVA shall fill the data in the variables `Door_Status_Left` and as follows:

5.10 Train and vehicle specific values

5.10.1 Maximum train speed

5.10.1.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`Max_Train_Speed`	TCMS to ATO	Sporadic, Triggered by ATO	Specific logic, see below

Table 34: Maximum available tractive effort functional vehicle adaptor interface

5.10.1.2 Rules / Sequence

API Message	From- To	Activation
Max_Available_Traction_Effort	FVA to ATO	Sent to ATO
P_Max_Train_Speed	Params to FVA	If P_Train_Data is set to static
[17] 6.2.7.1 Table 7 Maximum Train Speed	TCMS to FVA	If P_Train_Data is set to TCMS

Table 35: Maximum available tractive effort logic

5.10.2 Maximum available tractive effort (for the whole train)

5.10.2.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Max_Available_Traction_Effort	TCMS to ATO	Sporadic,	Specific logic, see below
Max_Available_Traction_Effort_unknown	TCMS to ATO	Sporadic,	Specific logic, see below

Table 36: Maximum available tractive effort functional vehicle adaptor interface

5.10.2.2 Rules / Sequence

API Message	From- To	Activation
Max_Available_Traction_Effort	FVA to ATO	Sent to ATO
Max_AvailTractionEffort_unknown	FVA to ATO	Sent to ATO
P_Max_AvailTractionEffort	Params to FVA	If P_Train_Data is set to static, then P_MaxAvailTractionEffort shall be used to set Max_Available_Traction_Effort. Note: if P_Max_AvailTractionEffort_unknown = true, then Max_Available_Traction_Effort shall be set to 0.
[17] 6.2.7.1 Table 7 Maximum available tractive effort (for the whole train)	TCMS to FVA	If P_Train_Data is set to TCMS then this variable shall be used to determin the API output values. If the variable has the special value for "unknown", then Max_AvailTractionEffort_unknown shall be set to "true" and Max_Available_Traction_Effort shall be set to 0. In all other cases, the value shall be output to Max_Available_Traction_Effort.

Table 37: Maximum available tractive effort logic

5.10.3 Maximum available tractive output power (for the whole train)

5.10.3.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Max_Available_Traction_Power	TCMS to ATO	Sporadic	Specific logic, see below
Max_Available_Traction_Power_unknown	TCMS to ATO	Sporadic	Specific logic, see below

Table 38: Maximum available tractive output power functional vehicle adaptor interface

5.10.3.2 Rules / Sequence

API Message	From- To	Activation
Max_Available_Traction_Power	FVA to ATO	Sent to ATO
Max_Available_Traction_Power_unknown	FVA to ATO	Sent to ATO
P_Max_AvailTractionPower	Params to FVA	If P_Train_Data is set to static, then P_MaxAvailTractionPower shall be used to set Max_Available_Traction_Power. Note: if P_Max_AvailTractionPower_unknown = true, then Max_Available_Traction_Power shall be set to 0.
[17] 6.2.7.1 Table 7 Maximum available tractive power (for the whole train)	TCMS to FVA	If P_Train_Data is set to TCMS then this variable shall be used to determin the API output values. If the variable has the special value for “unknown”, then Max_AvailTractionPower_unknown shall be set to “true” and Max_Available_Traction_Power shall be set to 0. In all other cases, the value shall be output to Max_Available_Traction_power.

Table 39: Maximum available tractive output power logic

5.10.4 Currently available tractive effort (for the whole train)

5.10.4.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Available_Traction_Effort	TCMS to ATO	Sporadic	Specific logic, see below

Available_Traction_Effort_unknown	TCMS to ATO	Sporadic	Specific logic, see below

Table 40: Currently available tractive effort functional vehicle adaptor interface

5.10.4.2 Rules / Sequence

API Message	From- To	Activation
Available_Traction_Effort	FVA to ATO	Sent to ATO
Available_Traction_Effort_unknown	FVA to ATO	Sent to ATO
P_Available_Traction_Effort	Params to FVA	If P_Train_Data is set to static, then P_Available_Traction_Effort shall be used to set Max_Available_Traction_Power. Note: if P_Available_Traction_Effort _unknown = true, then Available_Traction_Effort shall be set to 0.
[17] 6.2.7.1 Table 7 Currently available tractive effort (for the whole train)	TCMS to FVA	If P_Train_Data is set to TCMS then this variable shall be used to determin the API output values. If the variable has the special value for "unknown", then Available_Traction_Effort_unknown shall be set to "true" and Available_Traction_Effort shall be set to 0. In all other cases, the value shall be output to Available_Traction_Effort.

Table 41: Currently available tractive effort logic

5.10.5 Maximum available dynamic brake effort (for the whole train)

5.10.5.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Max_Available_DynamicBrake_Effort	TCMS to ATO	Sporadic,	Specific logic, see below
Max_Available_DynamicBrake_Effort_unknown	TCMS to ATO	Sporadic,	Specific logic, see below

Table 42: Maximum available dynamic brake effort functional vehicle adaptor interface

5.10.5.2 Rules / Sequence

API Message	From- To	Activation
Max_Available_DynamicBrake_Effort	FVA to ATO	Sent to ATO

Max_Available_DynamicBrake_Effort_unknown	FVA to ATO	Sent to ATO
P_Max_Available_DynamicBrake	Params to FVA	If P_Train_Data is set to static, then P_Max_Available_DynamicBrake shall be used to set Max_Available_Traction_Power. Note: if P_Max_Available_DynamicBrake_Effort_unknown = true, then Max_Available_DynamicBrake_Effort shall be set to 0.
[17] 6.2.7.1 Table 7 Maximum available dynamic brake effort (for the whole train)	TCMS to FVA	If P_Train_Data is set to TCMS then this variable shall be used to determin the API output values. If the variable has the special value for "unknown", then P_Max_Available_DynamicBrake_Effort_unknown shall be set to "true" and Max_Available_DynamicBrake_Effort shall be set to 0. In all other cases, the value shall be output to Max_Available_DynamicBrake_Effort.

Table 43: Maximum available dynamic brake effort logic

5.10.6 Maximum available dynamic brake power (for the whole train)

5.10.6.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Max_Available_DynamicBrake_Power	TCMS to ATO	Sporadic,	Specific logic, see below
Max_Available_DynamicBrake_Power_unknown	TCMS to ATO	Sporadic,	Specific logic, see below

Table 44: Maximum available dynamic brake power functional vehicle adaptor interface

5.10.6.2 Rules / Sequence

API Message	From- To	Activation
Max_Available_DynamicBrake_Power	FVA to ATO	Sent to ATO
Max_Available_DynamicBrake_Power_unknown	FVA to ATO	Sent to ATO
P_Max_Available_DynamicBrake_Power	Params to FVA	If P_Train_Data is set to static, then P_Max_Available_DynamicBrake_Power shall be used to set Max_Available_Traction_Power. Note: if P_Max_Available_DynamicBrake_Power_unknown = true, then Max_Available_DynamicBrake_Power

		shall be set to 0.
[17] 6.2.7.1 Table 7 Maximum available dynamic brake power unknown (for the whole train)	TCMS to FVA	If P_Train_Data is set to TCMS then this variable shall be used to determin the API output values. If the variable has the special value for "unknown", then P_Max_Available_DynamicBrake_Power_unknown shall be set to "true" and Max_Available_DynamicBrake_Power shall be set to 0. In all other cases, the value shall be output to Max_Available_DynamicBrake_Power.

Table 45: Maximum available dynamic brake power logic

5.10.7 Currently available dynamic brake effort (for the whole train)

5.10.7.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Available_DynamicBrake_Effort	TCMS to ATO	Sporadic,	Specific logic, see below
Available_DynamicBrake_Effort_unknown	TCMS to ATO	Sporadic,	Specific logic, see below

Table 46: Currently available dynamic brake effort functional vehicle adaptor interface

5.10.7.2 Rules / Sequence

API Message	From- To	Activation
Available_DynamicBrake_Effort	FVA to ATO	Sent to ATO
Available_DynamicBrake_Effort_unknown	FVA to ATO	Sent to ATO
P_Available_DynamicBrake_Effort	Params to FVA	If P_Train_Data is set to static, then P_Available_DynamicBrake_Effort shall be used to set Max_Available_Traction_Power. Note: if P_Available_DynamicBrake_Effort_unknown = true, then Available_DynamicBrake_Effort shall be set to 0.
[17] 6.2.7.1 Table 7 Currently available dynamic brake effort unknown (for the whole train)	TCMS to FVA	If P_Train_Data is set to TCMS then this variable shall be used to determin the API output values. If the variable has the special value for "unknown", then P_Available_DynamicBrake_Effort_unknown shall be set to "true" and Available_DynamicBrake_Effort shall be set to 0. In all other cases, the

		value shall be output to Available_DynamicBrake_Effort.

Table 47: Currently available dynamic brake effort logic

5.10.8 Train mass

5.10.8.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Train_Mass	TCMS to ATO	Sporadic	Specific logic, see below
Train_Mass_unknown	TCMS to ATO	Sporadic	Specific logic, see below

Table 48: Train mass functional vehicle adaptor interface

5.10.8.2 Rules / Sequence

API Message	From- To	Activation
Train_Mass	FVA to ATO	Sent to ATO
Train_Mass_unknown	FVA to ATO	Sent to ATO
P_Train_Mass	Params to FVA	If P_Train_Data is set to static, then P_Train_Mass shall be used to set Train_Mass. Note: if P_Train_Mass_unknown = true, then Train_Mass shall be set to 0.
[17] 6.2.7.1 Table 7 Train mass	TCMS to FVA	If P_Train_Data is set to TCMS then this variable shall be used to determine the API output values. If the variable has the special value for “unknown”, then P_Train_Mass_unknown shall be set to “true” and Train_Mass shall be set to 0. In all other cases, the value shall be output to Train_Mass.

Table 49: Train mass logic

5.10.9 T/B lever position

5.10.9.1 Overview

API Message	From- To	Sequence/ Activation	Rules
TB_Lever	TCMS to ATO	Cyclic	Passed through

Table 50: T/B lever position functional vehicle adaptor interface

5.10.1 T/B lever failure

5.10.1.1 Overview

API Message	From- To	Sequence/ Activation	Rules
TB_Lever	TCMS to ATO	Sporadic	Passed through

Table 51: T/B lever failure functional vehicle adaptor interface

5.10.2 Adhesion factor reduction

5.10.2.1 Overview

API Message	From- To	Sequence/ Activation	Rules
AdhesionFactor_Reduction	TCMS to ATO	Sporadic	Specific logic, see below

Table 52: Adhesion factor reduction functional vehicle adaptor interface

5.10.2.2 Rules / Sequence

API Message	From- To	Activation
AdhesionFactor_Reduction	FVA to ATO	Sent to ATO
P_AdhesionFactor_Reduction	Params to FVA	If P_Train_Data is set to static, then P_AdhesionFactor_Reduction shall be used to set Train_Mass.
[17] 6.2.7.1 Table 7 Adhesion factor reduction	TCMS to FVA	If P_Train_Data is set to TCMS then this variable shall be used to determine the API output values.

Table 53: Adhesion factor logic

5.10.3 Actual input current

5.10.3.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Actual_InputCurrent	TCMS to ATO	Sporadic	Specific logic, see below

Table 54: Actual input current functional vehicle adaptor interface

5.10.3.2 Rules / Sequence

API Message	From- To	Activation
Actual_InputCurrent	FVA to ATO	Sent to ATO
P_Actual_InputCurrent	Params to FVA	If P_Train_Data is set to static, then P_Actual_InputCurrent shall be used to set Actual_InputCurrent.
[17] 6.2.7.1 Table 7 Actual input current	TCMS to FVA	If P_Train_Data is set to TCMS then this variable shall be used to determine the API output values.

Table 55: Actual input current logic

5.10.4 T/B set value

5.10.4.1 Overview

API Message	From- To	Sequence/ Activation	Rules
TB_Set	TCMS to ATO	Sporadic	Specific logic, see below

Table 56: T/B set value functional vehicle adaptor interface

5.10.4.2 Rules / Sequence

API Message	From- To	Activation
TB_Set	FVA to ATO	Sent to ATO
P_TB_Set	Params to FVA	If P_Train_Data is set to static, then P_TB_Set shall be used to set TB_Set.
[17] 6.2.7.1 Table 7 T/B set value	TCMS to FVA	If P_Train_Data is set to TCMS then this variable shall be used to determine the API output values.

Table 57: T/B set value

5.10.5 Brake mode

5.10.5.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Brake_Mode	TCMS to ATO	Sporadic	Specific logic, see below

Table 58: Brake mode functional vehicle adaptor interface

5.10.5.2 Rules / Sequence

API Message	From- To	Activation
Brake_Mode	FVA to ATO	Sent to ATO
P_Brake_Mode	Params to FVA	If P_Train_Data is set to static, then P_Brake_Mode shall be used to set Brake_Mode.
[17] 6.2.7.1 Table 7 Brake mode	TCMS to FVA	If P_Train_Data is set to TCMS then this variable shall be used to determine the API output values.

Table 59: Brake mode logic

5.10.6 Wheel diameters

5.10.6.1 Overview

API Message	From- To	Sequence/ Activation	Rules
Wheel_Diameter	TCMS to ATO	Sporadic	Specific logic, see below
Wheel_Diameter_not_used	TCMS to ATO	Sporadic	Specific logic, see below

Table 60: Wheel diameters functional vehicle adaptor interface

5.10.6.2 Rules / Sequence

API Message	From- To	Activation
Wheel_Diameter	FVA to ATO	Sent to ATO
Wheel_Diameter_not_used	FVA to ATO	Sent to ATO
P_Wheel_Diameter	Params to FVA	If P_Train_Data is set to static, then P_Wheel_Diameter and P_Wheel_Diamater_not_used shall be used to set the related variables.
P_Wheel_Diameter_not_used		If P_Wheel_Diameter_not_used is true, then Wheel_Diamater_not_used shall be set to true and Wheel_Diameter shall be set to 0.
[17] 6.2.7.1 Table 7 Brake mode	TCMS to FVA	If P_Train_Data is set to TCMS then this variable shall be used to determine the API output values.

Table 61: Wheel diameters logic

5.11 TCMS Capability

5.11.1 TCMS Capability Request

5.11.1.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`TCMS_capabilities_request`	ATO to FVA	Sporadic, Triggered by ATO	Processed by FVA

Table 62: TCMS Capability Request functional vehicle adaptor interface

5.11.2 TCMS Capability Report

5.11.2.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`TCMS_capabilities_report`	FVA to ATO	Sporadic, Triggered by ATO	Generated by FVA based on parameters

Table 63: TCMS Capability report functional vehicle adaptor interface

5.12 Time

5.12.1 UTC time

5.12.1.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`UTC_Time`	TCMS to ATO	Cyclic, at least every 10ms	Specific logic, see below

Table 64: UTC time functional vehicle adaptor interface

5.12.2 UTC date

5.12.2.1 Overview

API Message	From- To	Sequence/ Activation	Rules
`UTC_Date`	TCMS to ATO	Cyclic, at least every 10ms	Specific logic, see below

Table 65: UTC date functional vehicle adaptor interface

5.12.2.2 Rules / Sequence

API Message	From- To	Activation
UTC_Time	FVA to ATO	Sent to ATO
UTC_Date	FVA to ATO	Sent to ATO
P_UTC_Time_and_Date	Params to FVA	Used to parameterize the FVA function Manage_Date_and_Time
[17] 6.2.8.1 UTC time	TCMS to FVA	If P_UTC_Time_and_Date is set to TCMS then this variable shall be used to determine the API output values. The variable is then used by the function Manage_Date_and_Time
UTC_Time_and_Date_LL	TCMS to ATO	If P_UTC_Time_and_Date is set to TCMS_advanced then this variable shall be used to determine the API output values.
		If P_UTC_Time_and_Date is set to FVA, then the FVA shall provide the Time and Date on its own.

Table 66: UTC time and date logic

5.12.3 Time and date (low- level)

5.12.3.1 A FVA function shall be created that formats UTC time and date according to the specification of UTC_time and UTC_date.

Function	Interface	Description
Manage_Date_and_Time		
	UTC_time	Output to ATO
	UTC_date	Output to ATO
	[17] 6.2.8.1 UTC Time	Input from TCMS (if appropriate)
	UTC_Time_and_Date_LL	Input from TCMS (if appropriate)
	P_UTC_Time_and_Date	Parameters
Functional Description	Project- specific function providing UTC time and date	

Table 67: Time and date low- level function

5.13 Brake and Traction Models

5.13.1 Brake and Traction models

5.13.1.1 It shall be possible to send the brake and traction models to the ATO

5.13.1.2 If the TCMS provides such data, they shall be forwarded

5.13.1.3 In other cases, the FVA shall send the statically saved parameters to the ATO

5.13.1.4 Rules / Sequence

API Message	From- To	Activation
Brake_and_Traction_Models	FVA to ATO	Sent to ATO
P_Brake_and_Traction_Models	FVA to ATO	If P_Mode_Brake_and_Traction_Models is set to FVA then this variable shall be used to determine the API output values.
Brake_and_Traction_Models_LL	TCMS to ATO	If P_Mode_Brake_and_Traction_Modelsis set to TCMS_advanced then this variable shall be used to determine the API output values.

Table 68: Brake and Traction model logic

6 Additional/ Optional Data exchanged with TCMS (in addition to Subset-139)

6.1 Variables

6.1.1 AD_BINARY_ENGAGE_DIRECT_BRAKE_ACPU

Name	**AD_BINARY_ENGAGE_DIRECT_BRAKE_ACPU**		
Description	Binary engage indirect brake cmd Note: when this signal is set to 0, the brake will usually maintain the current pressure. The exact implementation of this signal is application- specific.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not engage direct brake	
	1	Engage direct Brake	

6.1.2 AD_BINARY_ENGAGE_INDIRECT_BRAKE_ACPU

Name	**AD_BINARY_ENGAGE_INDIRECT_BRAKE_ACPU**		
Description	Binary engage indirect brake cmd Note: when this signal is set to 0, the brake will usually maintain the current pressure. The exact implementation of this signal is application- specific.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not engage indirect brake	
	1	Engage Indirect Brake	

6.1.3 AD_BINARY_LOW_PRESSURE_OVERFILLING_ACPU

Name	**AD_BINARY_LOW_PRESSURE_OVERFILLING_ACPU**		
Description	Used for quick brake emulation		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not apply low pressure overfilling	
	1	apply low pressure overfilling	

6.1.4 AD_BINARY_RELEASE_DIRECT_BRAKE_ACPU

Name	**AD_BINARY_RELEASE_DIRECT_BRAKE_ACPU**		
Description	Binary release direct brake cmd Low- level control of direct brake Note: when this signal is set to 0, the brake will usually maintain the current pressure. The exact implementation of this signal is application- specific.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not release direct brake	
	1	Release direct Brake	

6.1.5 AD_BINARY_RELEASE_INDIRECT_BRAKE_ACPU

Name	**AD_BINARY_RELEASE_INDIRECT_BRAKE_ACPU**		
Description	Binary release indirect brake cmd Low- level control of indirect brake Note: when this signal is set to 0, the brake will usually maintain the current pressure. The exact implementation of this signal is application- specific.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not release indirect brake	
	1	Release Indirect Brake	

6.1.6 AD_BINARY_TRACTION_0_ACPU

Name	**AD_BINARY_TRACTION_0_ACPU**		
Description	Low- level control of traction Force traction to 0 Note: The behavior of this variable is project specific.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not Force traction to 0	
	1	Force traction to 0	

6.1.7 AD_BINARY_TRACTION_DOWN_ACPU

Name	**AD_BINARY_TRACTION_DOWN_ACPU**		
Description	Low- level control of traction True – Decrease Traction False – Do not decrease traction Note: The behavior of this variable is project specific.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not decrease traction	
	1	Decrease traction	

6.1.8 AD_BINARY_TRACTION_UP_ACPU

Name	**AD_BINARY_TRACTION_UP_ACPU**		
Description	Binary traction request – increase Low- level control of traction True – Increase Traction False – Do not increase traction Note: The behavior of this variable is project specific.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not increase traction	
	1	Increase traction	

6.1.9 AD_ACPU_HIGH_PRESSURE_FILLING_ACPU

Name	**AD_ACPU_HIGH_PRESSURE_FILLING_ACPU**

Description	High pressure filling, used for quick brake emulation		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not apply high pressure filling	
	1	Apply high pressure filling	

6.2 Packets FVA- TCMS

6.2.1 Packet 40: Brake model request

Packet ID	40		
Description	Brake Model Request		
Sent	Sporadically		
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	BRAKE_DELAY_CLASS_ID_ACPU	8	
	DECELERATION_CLASS_ID_ACPU	8	

6.2.2 Packet 41: Direct Traction / Brake Commands

Packet ID	41		
Description	Commands to directly control the brake and traction actuators without TCMS regulation		
Sent	Sporadically		
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	AD_BINARY_RELEASE_INDIRECT_BRAKE_ACPU	1	
	AD_BINARY_ENGAGE_INDIRECT_BRAKE_ACPU	1	
	AD_BINARY_TRACTION_UP_ACPU	1	
	AD_BINARY_TRACTION_DOWN_ACPU	1	
	AD_BINARY_TRACTION_0_ACPU	1	
	AD_BINARY_RELEASE_DIRECT_BRAKE_ACPU	1	
	AD_BINARY_ENGAGE_DIRECT_BRAKE_ACPU	1	
	AD_BINARY_LOW_PRESSURE_OVERFILLING_ACPU	1	
	AD_ACPU_HIGH_PRESSURE_FILLING_ACPU	1	

6.3 Packets TCMS- FVA

6.3.1.1 Packet 50: Brake models

Packet ID	50		
Description	Model of the emergency brake, traction, and service brake (if present), to be used by the Core CPU		
Sent	Sporadically		
Content	***Variable***	***Length***	***Comment***
	NID_PACKET	8	
	L_PACKET	13	
	Q_BRAKE_MODEL_TCMS	1	The following fields only if Q_BRAKE_MODEL = 1
	MODEL_BEGIN_BRAKE_TCMS	8	Part of EB model
	MODEL_FULL_BRAKE_TCMS	11	Part of EB model
	N_ITER	5	Part of EB model max. value: 5
	MODEL_SPEED_TCMS(k)	8	Part of EB model
	MODEL_DECELER_TCMS(k)	8	Part of EB model
	CUT_TRACT_DELAY_TCMS	8	Part of traction model
	TRAIN_MAX_ACC_TCMS	10	Part of traction model

Packet ID	50		
	ACC_COEF_SB_UNUSED_TCMS	2	Part of traction model
	ACC_COEF_SB_USED_TCMS	2	Part of traction model
	Q_SB_MODEL_PRESENT	1	Part of SB model
	MODEL_BEGIN_BRAKE_TCMS	8	Part of SB model
	MODEL_FULL_BRAKE_TCMS	11	Part of SB model
	N_ITER	5	Part of SB model max. value: 5
	MODEL_SPEED_TCMS(k)	8	Part of SB model
	MODEL_DECELER_TCMS(k)	8	Part of SB model
	MIN_ROT_MASS_PERCENT_TCMS	8	Part of rot mass model
	NOM_ROT_MASS_PERCENT_TCMS	8	Part of rot mass model
	MAX_ROT_MASS_PERCENT_TCMS	8	Part of rot mass model
	T_W_TCMS	13	Part of driver delay
	T_P_TCMS	13	Part of driver delay
	T_I_P_TCMS	13	Part of driver delay
	T_RSMA_TCMS	13	Part of driver delay

6.3.1.1.1 Packet 50 is sent sporadically by the TCMS.

6.4 External functions

6.4.1 Variables

6.4.1.1 BR_DISTRI_PRESS_X_TCMS

Name	**BR_DISTRI_PRESS_X_TCMS**		
Description	Pressure at brake distributor output		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 16 bit	0 mbar 0	10 000 mbar 10000	1 mbar
Special/ Reserved Values	0-65534	Spare	
	65535	Unknown	

6.4.1.2 BR_PIPE_PRESS_X_TCMS

Name	**BR_PIPE_PRESS_X_TCMS**		
Description	Brake pipe pressure		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 16 bit	0 mbar 0	10 000 mbar 10000	1 mbar
Special/ Reserved Values	0-65534	Spare	
	65535	Unknown	

6.4.1.3 BRAKE_LEVERS_POS_X_TCMS

Name	**BRAKE_LEVERS_POS_X_TCMS**		
Description	Brake levers position		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Unsigned Integer 8 bit			
Special/ Reserved Values	0	All brake levers in zero positions	
	1	Any of brake levers is out of neutral position	
	2-254	Spare	
	255	Unknown	

6.4.1.4 BRAKE_MODE_X_TCMS

Name	**BRAKE_MODE_X_TCMS**
Description	Brake mode

Name	**BRAKE_MODE_X_TCMS**		
Type	**_Minimum Value_**	**_Maximum Value_**	**_Resolution/ Formula_**
Unsigned Integer 8 bit			
Special/ Reserved Values	0	G	
	1	P - freight train	
	2	P - passenger train	
	3	R	
	4-255	Spare	

6.4.1.5 BRAKE_STATUS_X_TCMS

Name	**BRAKE_STATUS_X_TCMS**		
Description	Brake status - Auxiliary logical control signals for pneumatic brakes control Included FIS signals: EB released, SB applied, Holding Brake applied, Direct brake applied, Traction over brake enabled,		
Type	**_Minimum Value_**	**_Maximum Value_**	**_Resolution/ Formula_**
Unsigned Integer 8 bit			
Special/ Reserved Values	0b0000 0001	EBrel	
	0b0000 0010	SBapp	
	0b0000 0100	HBapp	
	0b0000 1000	DirBApp	
	0b0001 0000	Spare	
	0b0010 0000	brake cleaning / hill start	
	0b0100 0000	Spare	
	0b1000 0000	Spare	

6.4.1.6 DBS_X_ACPU

Name	**DBS_X_ACPU**		
Description	Activate external function EXT_DynamicBrakeStatus		
Type	**_Minimum Value_**	**_Maximum Value_**	**_Resolution/ Formula_**
Boolean 1 bit			
Special/ Reserved Values	0	Do not activate external function EXT_DynamicBrakeStatus	
	1	Activate external function EXT_DynamicBrakeStatus	

6.4.1.7 ENGAGEMENT_READY_EXT

Name	**ENGAGEMENT_READY_EXT**		
Description	Engagement ready Explanation: All conditions for engagement are fulfilled (including door closed, direction selected, etc.). If this signal disappears, ATO disengages. When the signal re-appears, driver must push engage button for continuing in automated mode.		
Type	**_Minimum Value_**	**_Maximum Value_**	**_Resolution/ Formula_**
Boolean 1 bit			
Special/ Reserved Values	0	Engagement not ready	
	1	Engagement ready	

6.4.1.8 INDIRECT_BRAKE_ENGAGE_BIN_X_EXT

Name	**INDIRECT_BRAKE_ENGAGE_BIN_X_EXT**		
Description	Binary Engagement of Indirect Brake Low- Level control of Indirect Brake Variable received from external function		
Type	**_Minimum Value_**	**_Maximum Value_**	**_Resolution/ Formula_**

Name	INDIRECT_BRAKE_ENGAGE_BIN_X_EXT		
Boolean 1 bit			
Special/ Reserved Values	0	Do Not Engage Indirect Brake	
	1	Engage Indirect Brake	

6.4.1.9 PBR_X_ACPU

Name	PBR_X_ACPU		
Description	Activate external function EXT_PneumaticBrakeRequest		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not activate external function EXT_PneumaticBrakeRequest	
	1	Activate external function EXT_PneumaticBrakeRequest	

6.4.1.10 RELATIVE_TRACTION_REQUEST_X_ACPU

Name	RELATIVE_TRACTION_REQUEST_X_ACPU		
Description	Percentage of traction/brake capability of the train. Range: -100% (full brake) ... 0 ... +100% (full traction), resolution ≤ 0.1% In order to achieve the required precision with integer value, this value is coded with a scaling factor of 10: -100.0% is coded as -1000 100.0% is coded as 1000 Variable sent to external function		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 11 bit	-100.0% -1000	100.0% 1000	0,1%

6.4.1.11 RTR_X_ACPU

Name	RTR_X_ACPU		
Description	Activate external function EXT_RelativeTractionRequest		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not activate external function EXT_RelativeTractionRequest	
	1	Activate external function EXT_RelativeTractionRequest	

6.4.1.12 RTS_X_ACPU

Name	RTS_X_ACPU		
Description	Activate external function EXT_RelativeTractionStatus		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not activate external function EXT_RelativeTractionStatus	
	1	Activate external function EXT_RelativeTractionStatus	

6.4.1.13 INDIRECT_BRAKE_RELEASE_BIN_X_EXT

Name	**INDIRECT_BRAKE_RELEASE_BIN_X_EXT**		
Description	Binary Release of Indirect Brake Low- Level control of Indirect Brake Variable received from external function		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do Not Release Indirect Brake	
	1	Release of Indirect Brake	

6.4.1.14 TRACTION_APPLIED_EXT

Name	**TRACTION_APPLIED_EXT**		
Description	Traction applied Explanation: Propulsion reports that traction is applied		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Traction not applied	
	1	Traction applied	

6.4.1.15 TRACTION_UP_BIN_X_EXT

Name	**TRACTION_UP_BIN_X_EXT**		
Description	Binary traction request – increase Low- level control of traction True – Increase Traction False – Do not increase traction Note: The behavior of this variable is project specific. Variable received from external function		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not increase traction	
	1	Increase traction	

6.4.1.16 TRACTION_DOWN_BIN_X_EXT

Name	**TRACTION_DOWN_BIN_X_EXT**		
Description	Low- level control of traction True – Decrease Traction False – Do not decrease traction Note: The behavior of this variable is project specific. Variable received from external function		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not decrease traction	
	1	Decrease traction	

6.4.1.17 TRACTION_0_BIN_X_EXT

Name	**TRACTION_0_BIN_X_EXT**		
Description	Low- level control of traction Force traction to 0 Note: The behavior of this variable is project specific. Variable received from external function		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***

Name	**TRACTION_0_BIN_X_EXT**		
Boolean 1 bit			
Special/ Reserved Values	0	Do not Force traction to 0	
	1	Force traction to 0	

6.4.1.18 TRACTION_APPLIED_X_EXT

Name	**TRACTION_APPLIED_X_EXT**		
Description	Traction applied Explanation: Propulsion reports that traction is applied		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Traction not applied	
	1	Traction applied	

6.4.1.19 DIRECT_BRAKE_RELEASE_BIN_X_EXT

Name	**DIRECT_BRAKE_RELEASE_BIN_X_EXT**		
Description	Binary release direct brake cmd Low- level control of direct brake Note: when this signal is set to 0, the brake will usually maintain the current pressure. The exact implementation of this signal is application- specific. Variable received from external function		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not release direct brake	
	1	Release direct Brake	

6.4.1.20 DIRECT_BRAKE_ENGAGE_BIN_X_EXT

Name	**DIRECT_BRAKE_ENGAGE_BIN_X_EXT**		
Description	Binary engage direct brake cmd Note: when this signal is set to 0, the brake will usually maintain the current pressure. The exact implementation of this signal is application- specific. Variable received from external function		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not engage direct brake	
	1	Engage direct Brake	

6.4.1.21 DYNAMICBRAKEREQUEST_X_ACPU

Name	**DYNAMICBRAKEREQUEST_X_ACPU**		
Description	Control signal for direct control of dynamic brake by external function Range: 0 ... 100% (full direct brake), resolution ≤ 0.1%		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 10 bit	0% 0	100.0% 1000	0,1%

6.4.1.22 DYNAMICBRAKEREQUEST_X_EXT

Name	**DYNAMICBRAKEREQUEST_X_EXT**		
Description	Control signal for direct control of dynamic brake by external function Range: 0 ... 100% (full direct brake), resolution ≤ 1%		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 10 bit	-100.0% -1000	100.0% 1000	0,1%

6.4.1.23 DYNAMIC_BRAKE_APPLIED_X_EXT

Name	**DYNAMIC_BRAKE_APPLIED_X_EXT**		
Description	Dynamic brake applied Propulsion reports that dynamic brake is applied. Managed by external function For Locos and EMUs only.		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Dynamic brake not applied	
	1	Dynamic brake applied	

6.4.1.24 DYNAMIC_BRAKE_READY_X_EXT

Name	**DYNAMIC_BRAKE_READY_X_EXT**		
Description	Dynamic brake ready All conditions for applying the dynamic brake are fulfilled. If this signal is active, then ATO-OB is allowed to request the dynamic brake. For Locos and EMU only. Managed by external function Note: This signal stays false if no dynamic brake is installed		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Dynamic brake not ready	
	1	Dynamic brake ready	

6.4.1.25 LOW_PRESSURE_OVERFILLING_X_EXT

Name	**LOW_PRESSURE_OVERFILLING_X_EXT**		
Description	Used for quick brake emulation		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not apply low pressure overfilling	
	1	apply low pressure overfilling	

6.4.1.26 HIGH_PRESSURE_FILLING_X_EXT

Name	**HIGH_PRESSURE_FILLING_X_EXT**		
Description	High pressure filling, used for quick brake emulation		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not apply high pressure filling	
	1	Apply high pressure filling	

6.4.1.27 INDIRECT_BRAKE_REQUEST_X_ACPU

Name	**INDIRECT_BRAKE_REQUEST_X_ACPU**		
Description	Immediate indirect air brake request Auxiliary control signal for direct control of indirect (train) air brake Range: 0.0 % to 100.0 %. Resolution <= 0.1% Note: 0% of brake force typically equals a brake pipe pressure of 5.0 bar, 100% equals a brake pipe pressure of 3.5 bar		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 10 bit	-100.0% -1000	100.0% 1000	0,1%

6.4.1.28 DIRECT_BRAKE_REQUEST_X_ACPU

Name	**DIRECT_BRAKE_REQUEST_X_ACPU**		
Description	Immediate direct air brake request Auxiliary control signal for direct control of direct (Locomotive) air brake Range: 0 ... 100% (full direct brake), resolution ≤ 1%		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Integer 10 bit	-100.0% -1000	100.0% 1000	0,1%

6.4.1.29 QUICK_BRAKE_RELEASE_X_ACPU

Name	**QUICK_BRAKE_RELEASE_X_ACPU**		
Description	Quick brake release request Auxiliary signal for quick brake release (mandatory for Locos, optional for EMUs). The function will be handled in TCMS using Low pressure overfilling (Angleicher) and/or High-pressure filling stroke (Füllstoss)		
Type	***Minimum Value***	***Maximum Value***	***Resolution/ Formula***
Boolean 1 bit			
Special/ Reserved Values	0	Do not release the quick brake	
	1	Release the quick brake	

7 Appendix: Discussion of some specific usage scenarios

7.1.1 Introduction

This interface specification is intended to facilitate the integration of standardised ATO onboard systems with vehicles of various configurations, featuring a wide range of capablilites.
While some functions and the exchange of the related data must always be implemented, others might me optional.

The design of a specific ATO – vehicle integration should consider the following:

- The basic functionality that is required for the correct functioning of the ATO system.
- The capabilities and interfaces of the existing TCMS.
- The additional control and status signals and data that may be available on the vehicle.
- The packets and variables supported by the ATO onboard unit.

The Subset-139 FFFIS is intended to cover all required data.
On legacy vehicles, it is possible that functional gaps are discovered during the ATO / Vehicle Integration project.

The FVA with its various interfaces needs to be parameterized in order to ensure correct routing of the information between the ATO and the TCMS.
Additionally, it may be required to design and implement a Specific Vehicle Interface and project- specific external functions or functional subsystems.

Some possible scenarios are given in this section.
While the scenarios have been selected based on typical use cases, it should be noted that they are not intended to be exhaustive. Each ATO / Vehicle integration must be analysed and implemented on its own merits.

The FVA is intended to simplify and standardise the design and implementation of ATO interfaces for a wide range of vehicles and their command and control interface.

7.1.2 Usage scenario 1: Integration of fully Subset-139- compatible ATO and TCMS, with no need for additional interface

7.1.2.1 Data flow model

Figure 7: Data flow model for usage scenario 1 illustrates a use case, where the ATO and the TCMS both fully support the variable set as defined in Subset-139. No additional data are exchanged.

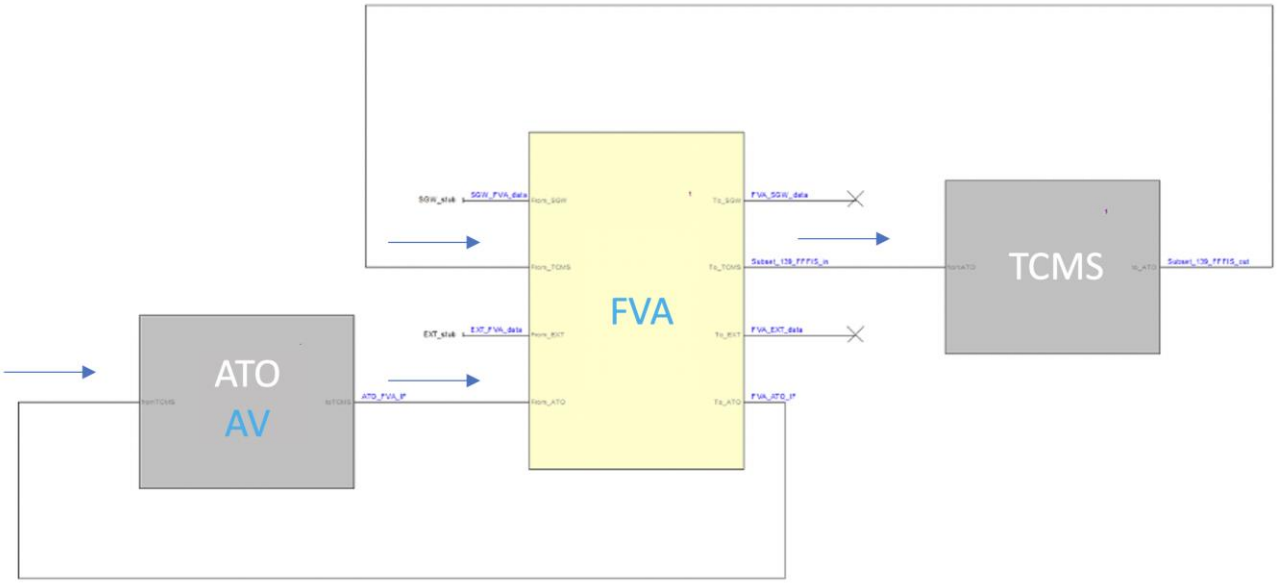


Figure 7: Data flow model for usage scenario 1

7.1.2.2 ATO

The ATO fully supports all variables that correspond to the variables as defined in the Subset-139 FIS.
This means that all standard functions can be covered by the system.
On each start up, the ATO requests the information on the capabilities of the connected TCMS/ FVA combination.
This way, correct functionality can be ensured even if the ATO onboard unit had to be replaced or updated.

7.1.2.3 FVA

The FVA is configured in a way that only the relevant packets and variables of the ATO Core interface are transmitted. The interfaces to the external function modules and to the SVI are deactivated, the related data flows are terminated by stubs.

7.1.2.4 Parameters

The parameter 5.2.35.1 P_STANDARD_139_CFG is set to 0 (The TCMS is fully Subset-139 compliant)

7.1.2.5 External Functions

No external functions are required

7.1.2.6 Specific Vehicle Interface

No SVI is required

7.1.2.7 TCMS

The TCMS is connected to the ATO via the FVA, using its standard Subset-139 FFFIS

7.1.3 Usage scenario 2: Integration of ATO and TCMS, with gaps in function and interface

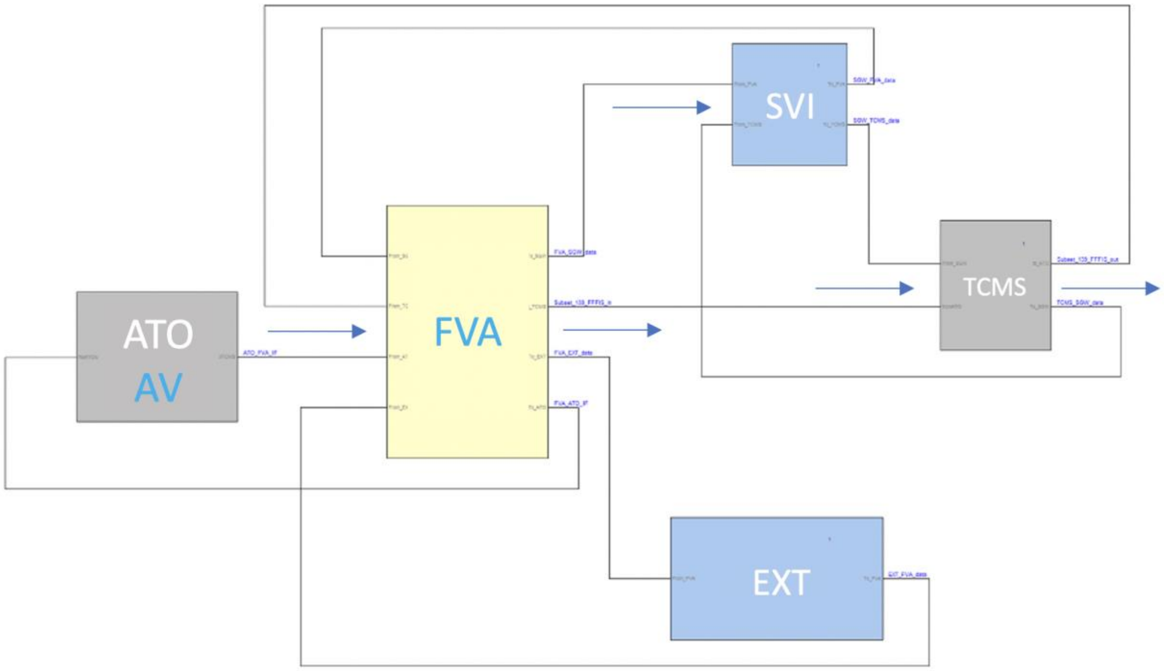


Figure 8:Data flow modekl for usage scenario 2

7.1.3.1 ATO

The ATO either fully supports the Subset-139 set of data or requires a superset.
In both cases, the full set of Subset-139 relevant variables are exchanged between the ATO onboard unit and the FVA.
Optionally, additional variables that are not known by the FFFIS might be availabe to the ATO through the FVA.
On each start up, the ATO requests the information on the capabilities of the connected TCMS/ FVA combination.
This way, correct functionality can be ensured even if the ATO onboard unit had to be replaced or updated.

7.1.3.2 FVA

Depending on the actual gap in functionality and / or data between the ATO and the TCMS, a certain set of data may not be directy forwarded from the ATO to the TCMS, but might serve as input values for certain external functions that could in turn drive alternate variables exchanged via the Specific Vehicle Interface with the vehicle.
A possible example could be a vehicle that has no high- level holding brake functionality implemented in the TCMS. In this case, the holiding brake functions and procedures could be implemented as external function, driving binary direct commands controlling pressure valves of the vehicle's pneumatic braking system.

7.1.3.3 Parameters

Depending on the actual gap in functionality and / or data between the ATO and the TCMS, a certain set of parameters has to be set by the project.
It is possible to fine- tune the routing for most variables, for example:

- Forward the value directly
- Call an external function
- Route the variable or a variable derived from an external function through the SVI

7.1.3.4 External Functions

Depending on the actual gap in functionality and / or data between the ATO and the TCMS, a certain set of external functions may be implemented, for example in order to implement the holding brake functions and procedures.

7.1.3.5 Specific Vehicle Interface

Depending on the actual gap in functionality and / or data between the ATO and the TCMS, a certain set of data may be exchanged between the TCMS and the FVA through the SVI.

7.1.3.6 TCMS

The TCMS exchanges packets/ variables with the FVA through both the FFFIS and the SVI. The actual set of variables routed through each of these interfaces is controlled by the set of parameters and is project- specific.

7.1.4 Usage scenario: Integration of fully OCORA compliant ATO and TCMS

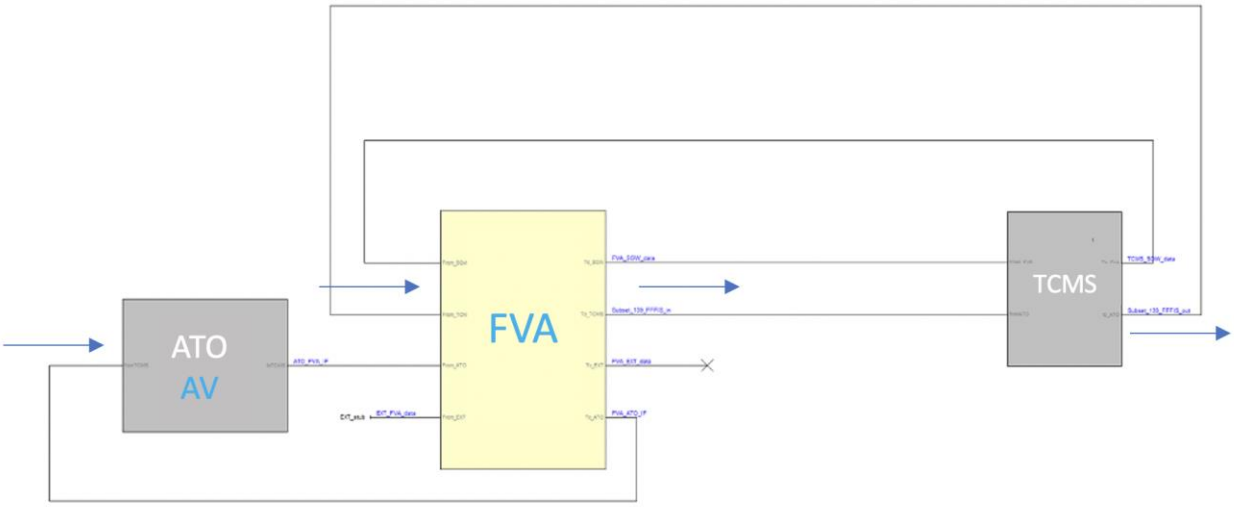


Figure 9: Fully OCORA compliant ATO and TCMS

7.1.4.1 ATO

The ATO supports the full OCORA set of packets/ variables, which are exchanged with the FVA.
On each start up, the ATO requests the information on the capabilities of the connected TCMS/ FVA combination.
This way, correct functionality can be ensured even if the ATO onboard unit had to be replaced or updated.

7.1.4.2 FVA

The FVA interfaces to the external functions are disabled, the related data flows are terminated and stubbed.
The SVI interface is directly connected to the TCMS, while the full set of packets/ variables as defined in Subset-139 [17] are exchanged with the TCMS through the FFFIS.

7.1.4.3 Parameters

The parameter 0

CPB_Full_Ocora is set to the value 1 (The TCMS is fully OCORA compliant).

7.1.4.4 External Functions

No external functions are implemented.

7.1.4.5 Specific Vehicle Interface

SVI data are directly exchanged with the TCMS.

7.1.4.6 TCMS

The TCMS supports both the Subset-139 [17] / 143 [26] and the SVI directly.

7.1.5 Usage scenario: Integration of ATO and legacy vehicle with no or partial TCMS

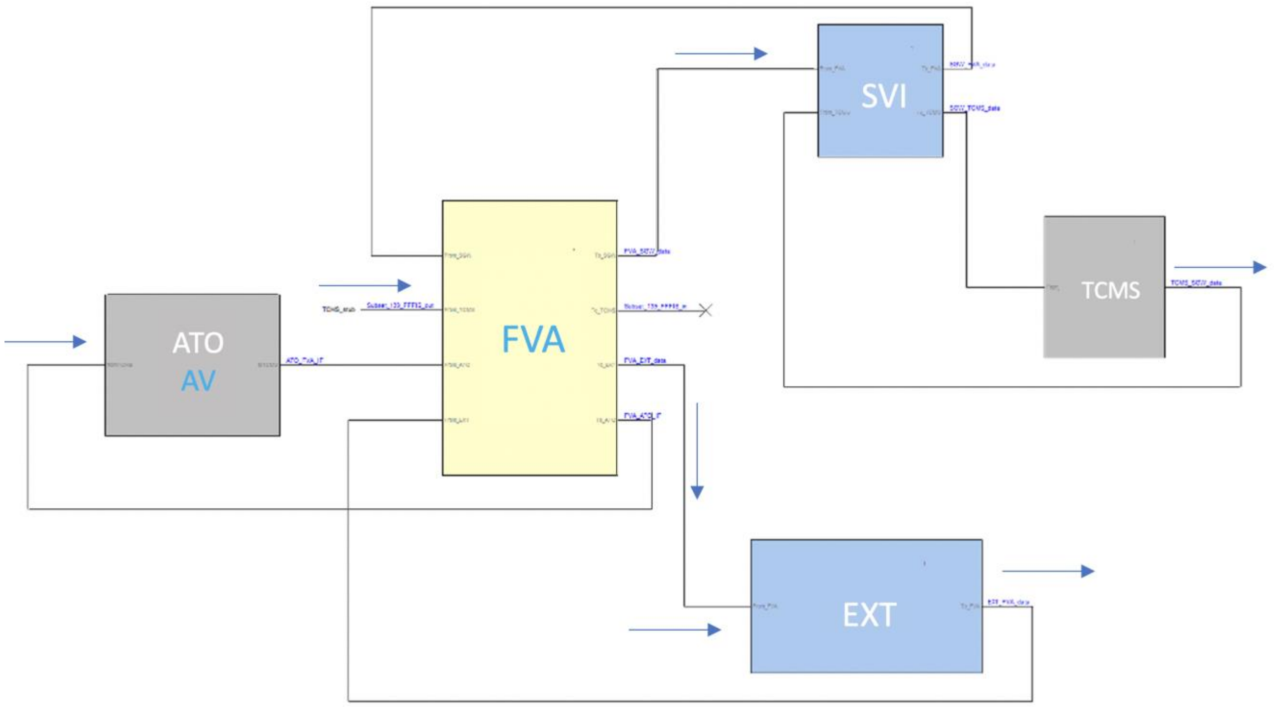


Figure 10: Usage scenario with no or only partial TCMS

7.1.5.1 ATO

The ATO either fully supports the Subset-139/ 143 set of data or requires a superset.
In both cases, the full set of Subset-139/ 143 relevant variables are exchanged between the ATO onboard unit and the FVA. Optionally, additional variables that are not known by the FFFIS might be availabe to the ATO through the FVA.
On each start up, the ATO requests the information on the capabilities of the connected TCMS/ FVA combination.
This way, correct functionality can be ensured even if the ATO onboard unit had to be replaced or updated.

7.1.5.2 FVA

As the TCMS has no possiblilty to (economically) have a Subset-139/ 143- compliant interface, the full functionality and the data exchange required for its implementation are realized through the SVI.
The actual details of the implementation of the SVI are project- specific .

7.1.5.3 Parameters

Depending on the actual gap in functionality and / or data between the ATO and the TCMS, a certain set of parameters has to be set by the project.
It is possible to fine- tune the routing for most variables, for example:

- Call an external function
- Route the variable or a variable derived from an external function directly through the SVI

7.1.5.4 External Functions

Depending on the actual gap in functionality and / or data between the ATO and the TCMS, a certain set of external functions may be implemented, for example in order to implement the holding brake functions and procedures.

7.1.5.5 Specific Vehicle Interface

All data are sent through the SVI

7.1.5.6 TCMS

The TCMS exchanges packets/ variables with the FVA through the SVI. The actual configuration is project- specific.