

**Research & Vehicle Technology**

**“Infotainment Systems Product Development”**

**Feature – Transport Protocol**

**APIM Infotainment Subsystem Part Specific Specification (SPSS)**

Version 1.7.1

**UNCONTROLLED COPY IF PRINTED**

**Version Date: April 5, 2016**

**FORD CONFIDENTIALF**

**Revision History**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Date** | | **Version** | | **Notes** | | | |
| **May 31, 2013** | | **1.0** | | **Initial Release** | | |  |
|  | |  | |  | | |  |
| **October 17, 2013** | | **1.1** | | **Updated Release** | | |  |
|  | | TP-GFEA-295336-2-Transport Protocol | | | | | asimukhi: added new signal ID 0x8D RadioText\_St to support 128 characters on DAB. |
|  | | TP-PHY-GTPC-138122-6-AUDIO - RDISP | | | | | asimukhi: added signals GetDABPresetInfo\_Rsp;GetDABStationList\_Rsp; EnsembleName2\_St; RadioText2\_St; GetDABEnsembleName\_Rsp; to fulfill the DAB-improvements. |
|  | | TP-PHY-GTPC-160781-2-MediaPlayerServer - MediaPlayerClient | | | | | asimukhi: added signal GetDABPresetInfo\_Rsp to fulfill the DAB-improvements. |
|  | | TP-LOG-GTPC-296321-1-SID-8D-RadioText2\_St | | | | | asimukhi: added support for up to 128 characters |
|  | |  | |  | | |  |
| **December 4, 2013** | | **1.2** | | **Updated Release** | | |  |
|  | | TP-GREQ-138092-7-Signal Utilization | | | | | sorris1: Added MobileCom\_Service2 for Embedded Modem |
|  | | TP-PHY-GTPC-223473-3-TCU - APIM | | | | | sorris1: Added SID 90 |
|  | | TP-LOG-GTPC-305875-1-SID-90-EmergencyCallText\_St | | | | | sorris1: New Requirement |
|  | |  | |  | | |  |
| **July 18, 2014** | | **1.3** | | **Updated Release** | | |  |
|  | | TP-FRD-REQ-023115/B-Transport Protocol (TcSE ROIN-295336-2) | | | | | rpaquet2 added new channels for APIM to AHUD and APIM to Rear EFP, added new Tp signal UpcomingStreetName\_St |
|  | | TP-REQ-015128/A-Signal Utilization (TcSE ROIN-138092-7) | | | | | sorris1: Added MobileCom\_Service2 for Embedded Modem |
|  | | STR-070474/B-Signal/Channel Mapping Tables (TcSE ROIN-295338) | | | | | rpaquet2 - Added new channels from APIM to AHUD/Rear EFP |
|  | | TP-PHY-TPP-REQ-023132/A-TCU - APIM (TcSE ROIN-223473-3) | | | | | sorris1: Added SID 90 |
|  | | TP-PHY-TPP-REQ-092284/A-NavRepServer - NavRepClient2 | | | | | new channel for APIM to AHUD |
|  | | TP-PHY-TPP-REQ-092285/A-NavRepClient2 - NavRepServer | | | | | New channel from AHUD to APIM |
|  | | TP-PHY-TPP-REQ-092286/A-MediaPlayerServer - MediaPlayerClient2 | | | | | new channel for APIM to AHUD |
|  | | TP-PHY-TPP-REQ-092287/A-MediaPlayerClient2 - MediaPlayerServer | | | | | New channel for AHUD to APIM |
|  | | TP-PHY-TPP-REQ-092288/A-PHONE - RDISP2 | | | | | New channel for APIM to AHUD |
|  | | TP-PHY-TPP-REQ-092289/A-RDISP2 - PHONE | | | | | new channel for AHUD to APIM |
|  | | TP-PHY-TPP-REQ-092294/A-MediaPlayerServer - MediaPlayerClient3 | | | | | new channel for APIM to Rear EFP |
|  | | TP-PHY-TPP-REQ-092295/A-MediaPlayerClient3 - MediaPlayerServer | | | | | New channel for Rear EFP to APIM |
|  | | STR-070475/B-Signal Descriptions (TcSE ROIN-295339) | | | | | rpaquet2 - Added UpcomingStreetName\_St |
|  | | TP-LOG-TPL-REQ-092298/A-SID-91-UpcomingStreetName\_St | | | | | New Requirement |
|  | |  | |  | | |  |
| **April 6, 2015** | | **1.4** | | **Updated Release** | | |  |
|  | | TP-PHY-TPP-REQ-023116/B-AUDIO - RDISP (TcSE ROIN-138122-6)+ | | | | | asimukhi: Message IDs 8A, 8B, 8C, 8E have been removed from the channel due to no need to be implemented. |
|  | | TP-PHY-TPP-REQ-023116/C-AUDIO - RDISP (TcSE ROIN-138122-6)+ | | | | | sberg15: added 0x76 LBP1\_ItemInfo\_Rsp and 0x76 MediaInformation\_St to the channel AUDIO-RDISP |
|  | | TP-PHY-TPP-REQ-023116/D-AUDIO - RDISP (TcSE ROIN-138122-6) | | | | | sberg15: added DynamicLabelPlus\_St and JournalineTxtMsg\_St signals for DAB |
|  | | TP-PHY-TPP-REQ-023124/B-NavRepServer - NavRepClient (TcSE ROIN-160780-1) | | | | | sberg15: added 0x76 LBP1\_ItemInfo\_Rsp to the channel NavRepserver-NavRepClient |
|  | | TP-PHY-TPP-REQ-023125/B-MediaPlayerServer - MediaPlayerClient (TcSE ROIN-160781-2) | | | | | asimukhi: Message ID 8A have been removed from the channel due to no need to be implemented. |
|  | | TP-PHY-TPP-REQ-023126/B-PHONE - MC (TcSE ROIN-160782-3) | | | | | sberg15: added 0x76 LBP1\_ItemInfo\_Rsp to the channel Phone-MC. |
|  | | TP-PHY-TPP-REQ-023135/B-CD - RDISP (TcSE ROIN-206152-1) | | | | | sberg15: added 0x76 LBP1\_ItemInfo\_Rsp and 0x79 MediaInformation\_St to the channel CD-MC |
|  | | TP-LOG-TPL-REQ-023173/B-SID-77-Destination\_Info\_St (TcSE ROIN-160691-3) | | | | | rpaquet2 - Added vector information back into the vector section of this TP method. |
|  | |  | |  | | |  |
| **October 7, 2015** | | **1.5** | | **Updated Release** | | |  |
|  | | TP-PHY-TPP-REQ-023117/B-SDARS - RDISP (TcSE ROIN-147073-4) | | | | | rpaquet2 - Added 0x76 to the list. |
|  | | TP-PHY-TPP-REQ-013860/B-TMCServer - TMCClient (TcSE ROIN-159708-3) | | | | | sberg15: added signal ID 0x99 TrafficServiceProvider\_St |
|  | | TP-PHY-TPP-REQ-023128/B-TMCClient - TMCServer (TcSE ROIN-178758-2) | | | | | sberg15: added signal ID 0x9A TrafficeGetServiceProvider\_Rq |
|  | | TP-PHY-TPP-REQ-023131/B-APIM - TCU (TcSE ROIN-223472-2) | | | | | rpaquet2 - Added 0x94. |
|  | | TP-PHY-TPP-REQ-023132/B-TCU - APIM (TcSE ROIN-223473-3) | | | | | rpaquet2 - Added 0x95 through 0x98 for Wifi Hotspot feature. |
|  | | STR-070475/C-Signal Descriptions (TcSE ROIN-295339)+ | | | | | MBORREL4: Added Signal ID's 0x94-0x98 |
|  | | STR-070475/D-Signal Descriptions (TcSE ROIN-295339) | | | | | sberg15: added signal IDs 0c99 and 0x9A for traffic services tuner. |
|  | | TP-LOG-TPL-REQ-023181/B-SID-82-ChargeProfileList\_Rq (TcSE ROIN-223468-1) | | | | | wstephe1: Additional instructional notes for RspCode = MODIFY on Byte 5: NumberOfItems and Byte 6: StartIndex for clarification |
|  | | TP-LOG-TPL-REQ-166128/A-SID-94-WifiInfo\_Rq | | | | | rpaquet2 - Added new for Wifi Hotspot feature. |
|  | | TP-LOG-TPL-REQ-166129/A-SID-95-WifiInfo\_Rsp | | | | | rpaquet2 - Added new for Wifi Hotspot feature. |
|  | | TP-LOG-TPL-REQ-166130/A-SID-96-CarrierInfo\_Rsp | | | | | rpaquet2 - Added new for Wifi Hotspot feature. |
|  | | TP-LOG-TPL-REQ-166131/A-SID-97-DataUsage\_Rsp+ | | | | | rpaquet2 - Added new for Wifi Hotspot feature. |
|  | | TP-LOG-TPL-REQ-166131/B-SID-97-DataUsage\_Rsp | | | | | MBORREL4: Updated description for DataUsed & DataLeft. Added new parameters TotalData, DataUsedPercent, and DataUnits. Added Notes where applicable to explain Data values are to be in steps of 0.1 units |
|  | | TP-LOG-TPL-REQ-166132/A-SID-98-DeviceList\_Rsp+ | | | | | rpaquet2 - Added new for Wifi Hotspot feature. |
|  | | TP-LOG-TPL-REQ-166132/B-SID-98-DeviceList\_Rsp | | | | | MBORREL4: Updated to remove BlackList encoding |
|  | |  | |  | | |  |
| **December 18, 2015** | | **1.6** | | **Updated Release** | | |  |
|  | | TP-PHY-TPP-REQ-023118/B-RDISP - SDARS (TcSE ROIN-147074-2) | | | | | rpaquet2 - Added 0x6C to this channel as APIM will send when X40 SDARS is available. |
|  | | TP-PHY-TPP-REQ-023125/C-MediaPlayerServer - MediaPlayerClient (TcSE ROIN-160781-2) | | | | | rpaquet2 - Added 0x67, 0x68, 0x6C and 0x6F for X40 SDARS data coming from APIM now. |
|  | | TP-PHY-TPP-REQ-023132/C-TCU - APIM (TcSE ROIN-223473-3) | | | | | Added signalID (0xCF/FF) to channel TCU-APIM. |
|  | | TP-PHY-TPP-REQ-092286/B-MediaPlayerServer - MediaPlayerClient2 | | | | | rpaquet2 - Added 0x67, 0x6C and 0x6F for new X40 SDARS data now coming from the APIM |
|  | | TP-PHY-TPP-REQ-092294/B-MediaPlayerServer - MediaPlayerClient3 | | | | | rpaquet2 - Added 0x67, 0x68, 0x6C and 0x6F to this channel for X40 signals sent from APIM. |
|  | | STR-070475/E-Signal Descriptions (TcSE ROIN-295339) | | | | | Added logical signal (0xCF/FF) for megaTP (TP-on-TP) handling |
|  | | TP-LOG-TPL-REQ-023169/B-SID-76-LBP1\_ItemInfo\_Rsp (TcSE ROIN-159709-6) | | | | | sberg15: Updated utilization byte to show the utilization for different features like Phone, Media Player, navigation etc. |
|  | | TP-LOG-TPL-REQ-023181/B-SID-82-ChargeProfileList\_Rq (TcSE ROIN-223468-1) | | | | | wstephe1: Additional instructional notes for RspCode = MODIFY on Byte 5: NumberOfItems and Byte 6: StartIndex for clarification |
|  | | TP-LOG-TPL-REQ-166130/B-SID-96-CarrierInfo\_Rsp+ | | | | | MBORREL4: Removed MSISDN and updated signal to include two phone numbers (Ford & Lincoln). Updated data size as well. |
|  | | TP-LOG-TPL-REQ-166130/C-SID-96-CarrierInfo\_Rsp | | | | | MBORREL4: Updated to include Ford and Lincoln Landing URLs |
|  | | TP-LOG-TPL-REQ-166131/B-SID-97-DataUsage\_Rsp | | | | | MBORREL4: Updated all parameters to now include CounterHour/Minute/Second, PlanType, ExpiryRenewal Date/Month/Day/Year/Hour/Minute/Second, DataUsed & TotalData (to 2 decimal places), OverageFlag, DataPlanStatus. Added Invalid values to DataUsedUnits, TotalDataUnits, and OverageFlag |
|  | | TP-LOG-TPL-REQ-201616/A-SID-CF-megaTP\_ConsecutivePackage | | | | |  |
|  | | TP-LOG-TPL-REQ-201617/A-SID-FF-megaTP\_FirstPackage | | | | |  |
|  | |  | |  | | |  |
| **March 18, 2016** | | **1.7** | | **Updated Release** | | |  |
|  | | TP-REQ-015128/B-Signal Utilization (TcSE ROIN-138092-7) | | | | | kfent1: utilization 73 changed from TMC to Traffic Data |
|  | | TP-REQ-015129/B-Character Coding Flag (TcSE ROIN-138093-3) | | | | | tklein26: Renamed / refined "Coding Table III". Existing description unclear. Extended description with RawData. Added examples for RawData like TPEG traffic. |
|  | | TP-PHY-TPP-REQ-023117/C-SDARS - RDISP (TcSE ROIN-147073-4) | | | | | rpaquet2 - Added A2 to this channel for X40 |
|  | | TP-PHY-TPP-REQ-023118/C-RDISP - SDARS (TcSE ROIN-147074-2) | | | | | rpaquet2 - Added 0xA1 for SDARS X40. |
|  | | TP-PHY-TPP-REQ-023124/C-NavRepServer - NavRepClient (TcSE ROIN-160780-1) | | | | | rpaquet2 - Removed 0x76 LBP1\_ItemInfo\_Rsp from channel. |
|  | | TP-PHY-TPP-REQ-023126/C-PHONE - MC (TcSE ROIN-160782-3) | | | | | rpaquet2 - Removed 0x76 LBP1\_ItemInfo\_Rsp from this channel. |
|  | | TP-PHY-TPP-REQ-023131/C-APIM - TCU (TcSE ROIN-223472-2) | | | | | tklein26: Navigation interfaces added to support online traffic feature |
|  | | TP-PHY-TPP-REQ-023132/D-TCU - APIM (TcSE ROIN-223473-3) | | | | | kfent1: moved Signals CF/FF to dedicated Physical Channel  MBORREL4: Added TP for 9B - WifiHotspotMAC\_Rsp |
|  | | TP-PHY-TPP-REQ-207117/A-TRAFFIC - RDISP | | | | | tklein26: Initial Revision |
|  | | TP-PHY-TPP-REQ-207118/A-RDISP - TRAFFIC | | | | | tklein26: Initial Revision |
|  | | TP-PHY-TPP-REQ-207115/A-OPTIN - RDISP | | | | | tklein26: Initial Revision. Channel for opt-in interfaces e. g. from TCU to APIM/CHR |
|  | | TP-PHY-TPP-REQ-207116/A-RDISP - OPTIN | | | | | tklein26: Initial Revision. Channel for opt-in interfaces e. g. from APIM/CHR to TCU |
|  | | STR-070475/F-Signal Descriptions (TcSE ROIN-295339) | | | | | rpaquet2 - Added A1 and A2 for SDARS x40. MBORREL4: Added REQ-195173 |
|  | | TP-LOG-TPL-REQ-166129/B-SID-95-WifiInfo\_Rsp | | | | | MBORREL4: Updated Byte 5 range |
|  | | TP-LOG-TPL-REQ-166130/C-SID-96-CarrierInfo\_Rsp | | | | | MBORREL4: Updated to include Ford and Lincoln Landing URLs. Updated Byte 4 range |
|  | | TP-LOG-TPL-REQ-166131/C-SID-97-DataUsage\_Rsp | | | | | MBORREL4: Updated Data Size. Added UserID. |
|  | | TP-LOG-TPL-REQ-166132/C-SID-98-DeviceList\_Rsp | | | | | MBORREL4: Updated to reflect DeviceName change from 17 to 19 characters. Updated ListSize from 20 to 31. Updated Byte 7 range |
|  | | TP-LOG-TPL-REQ-207066/A-SID-9C-CCOISynchronizationSession\_Rq | | | | | tklein26: Initial revision |
|  | | TP-LOG-TPL-REQ-207067/A-SID-9D-CCOISynchronizationSettings\_Rsp | | | | | tklein26: Initial revision |
|  | | TP-LOG-TPL-REQ-207068/A-SID-9E-CCOISynchronizationAuthorizedUsers\_Rsp | | | | | tklein26: Initial revision |
|  | | TP-LOG-TPL-REQ-207069/A-SID-9F-CCOISynchronizationSummaryReport | | | | | tklein26: Initial revision |
|  | | TP-LOG-TPL-REQ-207070/A-SID-A0-CCOISettingsUpdate\_Rq | | | | | tklein26: Initial revision |
|  | | TP-LOG-TPL-REQ-207875/A-SID-A1-SDARS\_ChannelList\_Rsp | | | | | rpaquet2 - New Tp method for SDARS x40. |
|  | | TP-LOG-TPL-REQ-208270/A-SID-A2-SDARS\_ChannelList\_Rq | | | | | rpaquet2 - New Tp method for SDARS x40. |
|  | | TP-LOG-TPL-REQ-211456/A-SID-A5-CCOIUserPrompt\_Rq | | | | | tklein26: Initial revision |
|  | | TP-LOG-TPL-REQ-211457/A-SID-A6-CCOIUserPrompt\_Rsp | | | | | tklein26: Initial revision |
|  |  | |  | | |  | | |
| **April 5, 2016** | **1.7.1** | | | | **Updated Release** |  | | |
|  | TP-LOG-TPL-REQ-166131/B-SID-97-DataUsage\_Rsp | | | | | MBORREL4: Revert back to RevB for Bundle2 (UserID not yet added) | | |
|  | TP-LOG-TPL-REQ-166132/B-SID-98-DeviceList\_Rsp | | | | | MBORREL4: Revert back to RevB for Bundle2 (DeviceName and ListSize not yet increased) | | |

**Table of Contents**

[Revision History 2](#_Toc446341606)

[1 General Requirements 6](#_Toc446341607)

[1.1 Overview 6](#_Toc446341608)

[1.2 Transport Channels 6](#_Toc446341609)

[1.2.1 Logical Channel Layout 7](#_Toc446341610)

[1.2.2 TP-REQ-015127/A-Signal Indentifier (TcSE ROIN-138089-1) 8](#_Toc446341611)

[1.2.3 TP-REQ-015128/B-Signal Utilization (TcSE ROIN-138092-7) 8](#_Toc446341612)

[1.2.4 TP-REQ-015129/B-Character Coding Flag (TcSE ROIN-138093-3) 10](#_Toc446341613)

[1.2.5 TP-REQ-015130/A-RDS Latin Code Page (TcSE ROIN-169144-2) 11](#_Toc446341614)

[1.2.6 TP-REQ-015131/A-Setting Character Coding Flag (TcSE ROIN-146167-1) 11](#_Toc446341615)

[1.2.7 TP-REQ-015132/A-End of string definition (TcSE ROIN-146168-1) 11](#_Toc446341616)

[1.2.8 Dynamic Signal Length 11](#_Toc446341617)

[1.2.9 Command Execution Status 12](#_Toc446341618)

[1.3 Channel Management 19](#_Toc446341619)

[1.3.1 TP-REQ-015140/A-Concurrent Data Transmission (TcSE ROIN-145774-1) 19](#_Toc446341620)

[1.3.2 TP-REQ-015141/A-Multi-Channel Management (TcSE ROIN-199074-1) 19](#_Toc446341621)

[1.3.3 Signal Heartbeat 20](#_Toc446341622)

[1.4 Signal and Channel Catalog 21](#_Toc446341623)

[1.4.1 Signal/Channel Mapping Tables 21](#_Toc446341624)

[1.4.2 Signal Descriptions 34](#_Toc446341625)

[2 Appendix: Reference Documents 122](#_Toc446341626)

# General Requirements

## Overview

A standard CAN message can only transport 8 bytes of information from transmitter to receiver. For an application that needs to transmit a packet of information which is greater then 8 bytes requires the use of a network level service. This service manages the segmentation of the information packet into separate CAN frames and transports each frame over the network in sequential order to the receiver. The service on the receiver will then reassemble the original information packet from the frames and provide it to the receiving application.

The network level service used by Ford Motor Company Infotainment Systems is based upon the ISO 15765-2 Network Layers Services specification. Specifically, Ford Motor Company has directed the use of Vector CANtech Multi-channel Transport Protocol. Therefore, through out this document, the term "transport layer/transport protocol" has been used synonymously with "network layer".

The responsibility of the following sections is to add a higher level protocol on top of this layer which will be responsible for managing physical & logical channels, channel utilization, channel status, fault handling, and data encapsulation.

## Transport Channels

The network/transport layer services specified earlier establishes a communication link between two separate nodes. The link can be classified as a "physical channel" with a fixed assignment between two nodes to exchange data.

In other words, when two nodes wish to share data which requires the use of the "transport" protocol, the nodes must utilize the physical channels that have been assigned to them. The physical channel assignments are a requirement of the transport protocol and nodes can have multiple physical channels with other nodes.

Node A

Node B

Node C

Chn 1

Chn 2

Chn 3

Chn 5

Chn 6

Chn 4

In an effort to maximize the use of the physical channels between two nodes, the concept of logical signals within a fixed physical channel is introduced. The use of logical signals aids in assigning different types of information to be transported between nodes over the same assigned physical channel. For example, Node A has one physical channel (ID=CH\_ID) with Node B and can use the channel to send multiple types of information (Types 1, 2, 3). In the following example, each info type has been linked to a logical signal. The logical signals have then been linked to a physical channel.

($CH\_ID) Physical Channel

Logical Sig ($TP\_ID1)

Logical Sig ($TP\_ID2)

Logical Sig ($TP\_ID3)

Logical Sig ($TP\_ID)

Transport

Node A

Info Type 1

Info Type 2

Info Type 3

Node B

Transport

### Logical Channel Layout

As explained earlier, within a physical channel a logical signal is utilized to transport application level data from one node to another. The logical signal itself is represented as an information packet which has a header and associated data. The header contains information about the logical signal and the data in the logical signal. The App data portion of the packet is the logical signal specific data used by the application. A logical signal packet is structured with the following layout:

App

Data

[Byte 4-4095]

Char

Coding

[Byte 3]

CES

[Byte 2]

Signal

Utilization

[Byte 1]

Signal

ID

[Byte 0]

Header

Data

Logical Signal

Physical Channel

CAN ID

|  |  |
| --- | --- |
| **Field** | **Description** |
| Signal ID | The Signal Identifier is used to determine between the different logical signals. |
| Signal Utilization | The Signal Utilization is used to link the information in the signal to the service. |
| CES | The CES is to use to reflect the status of a response to a request. |
| Char Coding | The Character coding flag indicates the selected coding table for the App data. |
| App Data | Application specific data |

The logical signal header information is not listed explicitly in corresponding sequence diagrams and associated method tables of the SPSS.

**Note:** The logical signal packet structure defined above may not apply to all signals listed in the specification as some legacy signals may still be utilized. The legacy logical signals shall define their own signal packet structure within there respective sections.

#### TP-REQ-015124/A-Byte Definition (TcSE ROIN-138090-1)

All information within the channel is segmented in words, bytes and bits. The transmission shall begin with BYTE 0 and BYTE 0 shall always contain the Signal Identifier. The parsing of information shall begin at BYTE 0 and end with BYTE xyz.

#### TP-REQ-015125/A-Bit Definition (TcSE ROIN-138091-1)

Bit definition defines the bit position within the bytes.

#### TP-REQ-015126/A-Bit, Byte Ordering (TcSE ROIN-149367-3)

This section defines the order and significance of bits within bytes. Within a byte, the most significant bit, msb, is that which is transmitted first and the least significant bit, lsb, is that which is transmitted last, as illustrated below. The significance of the interior bits uniformly decreases in progression from msb to lsb. Therefore, Bit 0 is the msb and Bit 7 is the lsb. This representation follows "Motorola Sequential" format.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| msb |  | | | | | | lsb |
| Bit  0 | Bit 1 | Bit 2 | Bit 3 | Bit 4 | Bit 5 | Bit 6 | Bit 7 |
| Byte | | | | | | | |
|  |  |  |  |  |  |  |  |

The format above shall be applied to ordering segmented bit-fields (each less than 8 bits) within a byte.

For example:

Byte 3: Character Coding

Bit 0-5: reserved

Bit 6-7: Coding

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

If Coding Table II is selected, the byte/bit ordering would indicate : [00000001]

### TP-REQ-015127/A-Signal Indentifier (TcSE ROIN-138089-1)

The Signal Identifier is used to determine between the different logical transport channels within a connection. Upon reception of a channel packet, the signal ID shall be inspected by the receiver to determine the type of information contained in the channel data portion of the packet.

### TP-REQ-015128/B-Signal Utilization (TcSE ROIN-138092-7)

Since some logical signals can be reused for transporting similar type of information from different services (e.g. Folder\_Name from CD, Folder\_Name from USB), the Signal Utilization is used to link the information in the signal to the service.

Utilization assignments shall be done on an as needed basis.

|  |  |  |  |
| --- | --- | --- | --- |
| Utilization | | | |
| Device Group | Service Number | Parameter Name | Parameter Description |
| High nibble | Low nibble |
| 0 | 0 | Service\_Not\_Present | No service of category "Radio" present |
| 0 | 1 | Radio\_Service1 | AmFm Radio General  (AM, FM, AST) |
| 0 | 2 | Radio\_Service2 | SDARS |
| 0 | 3 | Radio\_Service3 | DAB |
| 0 | 4 | Radio\_Service4 | Dynamic Station List |
| 0 | 5 | Radio\_Service5 | Radio Tagging |
| 0 | 6 | Radio\_Sevice6 | HD Radio |
| 0 | 7-E | Radio\_Service{Reserved} | Radio Service {Reserved} |
| 0 | F | Radio\_ServiceInvalid | Service(s) invalid; inhibited |
| 1 | 0 | Service\_Not\_Present | No service of category "MediaPlayer" present |
| 1 | 1 | MP\_Media1 | CD |
| 1 | 2 | MP\_Media2 | BT Audio Streaming |
| 1 | 3 | MP\_Media3 | USB |
| 1 | 4 | MP\_Media4 | iPod |
| 1 | 5 | MP\_Media5 | SD |
| 1 | 6 | MP\_Media6 | DVD |
| 1 | 7 | MP\_Media7 | Generic Metadata |
| 1 | 8-E | MP\_Media{Reserved} | Media Player {Reserved} |
| 1 | F | MP\_ServiceInvalid | Service(s) invalid; inhibited |
| 2 | 0 | Service\_Not\_Present | No service of category "Navigation" present |
| 2 | 1 | Nav\_Service1 | Mobile navigation |
| 2 | 2 | Nav\_Service2 | Navigation |
| 2 | 3-E | Nav\_Service{Reserved} | Navigation Service {Reserved} |
| 2 | F | Nav\_ServiceInvalid | Service(s) invalid; inhibited |
| 3 | 0 | Service\_Not\_Present | No service of category "MobileCommunication" present |
| 3 | 1 | MobileCom\_Service1 | Mobile Phone |
| 3 | 2 | MobileCom\_Service2 | Embedded Modem |
| 3 | 3-E | MobileCom\_Service{Reserved} | Mobile communication Service {Reserved} |
| 3 | F | MobileCom\_ServiceInvalid | Service(s) invalid; inhibited |
| 4 | 0 | Service\_Not\_Present | No service of category "Voice" present |
| 4 | 1 | Voice\_Service1 | Voice Recognition |
| 4 | 2 | Voice\_Service2 | VR with text capturing |
| 4 | 3-E | Voice\_Service{Reserved} | Voice Recognition Service {Reserved} |
| 4 | F | Voice\_ServiceInvalid | Service(s) invalid; inhibited |
| 5 | 0 | Service\_Not\_Present | No service of category "Video" present |
| 5 | 1-E | Video\_Service{Reserved} | Video Service {Reserved} |
| 5 | F | Video\_ServiceInvalid | Service(s) invalid; inhibited |
| 6 | 0 | Service\_Not\_Present | No service of category "Office" present |
| 6 | 1-E | Office\_Service{Reserved} | Office Service {Reserved} |
| 6 | F | Office\_ServiceInvalid | Service(s) invalid; inhibited |
| 7 | 0 | Service\_Not\_Present | No service of category "Data" present |
| 7 | 1 | Data\_Service1 | SSP Data Services |
| 7 | 2 | Data\_Service2 | Component Diagnostic Data |
| 7 | 3 | Data\_Service3 | Traffic Data |
| 7 | 4 | Data\_Service4 | List Browser Data |
| 7 | 5 | Data\_Service5 | DataReport |
| 7 | 6-E | Data\_Service{Reserved} | Data Service {Reserved} |
| 7 | F | Data\_ServiceInvalid | Service(s) invalid; inhibited |
| 8 | 0 | Service\_Not\_Present | No service of category "ChargeProgramming" present |
| 8 | 1 | Charge\_Programming\_Service1 | Charge Programming |
| 8 | 2-E | Charge\_Programming\_Service{Reserved} | Charge Programming {Reserved} |
| 8 | F | Charge\_Programming\_ServiceInvalid | Service(s) invalid; inhibited |
| F | F | Invalid | General invalid |

### TP-REQ-015129/B-Character Coding Flag (TcSE ROIN-138093-3)

The Character coding field is used to represent the selected coding table for the trained data stream (for text based information) in the TP message.

**Byte 1: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I  
0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II  
0x00-0xFF Latin-9 (1 byte per char)

0x2: Coding Table III

0x00-0xFF RawData (Hexadecimal Notation)

If the character coding is set to "**Coding Table I**; *Unicode UTF-16*" than the 16-bit based ISO/IEC 10646 (Unicode); with the Unicode encoding form UTF-16BE (Big Endian) is to use.

If the character coding is set to "**Coding Table II**; *Latin-9*" then the single byte encoding form ISO-8859-15 (Latin-9) is to use.

If the character coding is set to state encoded values between 0x3 – 0x7 Reserved, than the 16-bit based ISO/IEC 10646 (Unicode); with the Unicode encoding form UTF-16BE (Big Endian) is to be used. See Coding Table I.

For all signals (e.g. GetPresetInfo2) with text content from RDS source (Radio Data System) the coding shall be fixed to RDS code-Table EBU Table1 specified within EN 50067. The Coding Table indicated in "Byte 1: Character Coding" shall be 0x1: Coding Table II.

For all signals with number content only (e.g. telephone numbers) the coding is fixed to Coding Table II if possible.

A coding table is not is applied to the embedded data in the TP signal if the character coding table is set to "**Coding Table III**; RawData”.

If it is required to detail also the format of the data then additional information (byte / word / dword / … or / coding / compression / …) is added to this parameter or to a feature specific documentation external to this document.

Example: (Format: word)

0x2: Coding Table III

0x0000-0xFFFF Hexidecimal Notation

Or (format: byte)

0x2: Coding Table III

0x00-0xFF Hexidecimal Notation

To allow a “late binding” and ensure upgradability it is allowed to remove the format/attribute information from the character coding and define this in an external feature specific document.

Example: (“late binding”)

0x2: Coding Table III

This results in the fact that this information is external / not visible to the TP layer and this section. These information is detailed in the feature specific documentation. This concept allows a modification of the data content later on without impacting the TP layer and this specification.

This is required for e. g. TPEG traffic data raw data stream, “file transfer” or customer opt-in data.

Also BCD coded data will use this coding table flag.

Example: (BCD)

0x2: Coding Table III

0x0-0xF Hexidecimal Notation

### TP-REQ-015130/A-RDS Latin Code Page (TcSE ROIN-169144-2)

Please refer to the RDS specification:

"IEC62106 Edition 2 see Annex E basic and extended RDS character sets"

### TP-REQ-015131/A-Setting Character Coding Flag (TcSE ROIN-146167-1)

Since many media interfaces can provide textual information in many different formats, the transmitting node shall be responsible for determining which coding table shall be utilized for the most accurate representation of the text based information. The character coding evaluation must be done for each text based message transfer.

**Character Code Determination for Coding Table I - Unicode Data**

The transmitter node must evaluate if at least one character of the pending data transfer (for text based information) could not be represented with the Coding table II. If at least one character of the pending data transfer (for text based information) cannot be represented with Coding Table II, the Character Coding Flag shall be set to Coding Table I. All text based information for the pending data transfer shall be encoded according to Coding table I.

**Character Code Determination for Coding Table II - Latin-9 Data**

The transmitter node must evaluate if all characters of the pending data transfer (for text based information) can be represented with Coding table II. If all characters of the pending data transfer (for text based information) can be represented with Coding Table II, the Character Coding Flag shall be set to Coding Table II. All text based information for the pending data transfer shall be encoded according to Coding table II.

### TP-REQ-015132/A-End of string definition (TcSE ROIN-146168-1)

For Coding Table I, the End Of String character (EOS), 0x0000 UTF-16BE, shall be used to terminate every string. If an empty string has to be transmitted, the End Of String character must be sent.

For Coding Table II, the End Of String character (EOS )is 0x00. If an empty string has to be transmitted, the End Of String character must be sent.

### Dynamic Signal Length

#### TP-REQ-015133/A-Dynamic Signal Definition (TcSE ROIN-146172-1)

All logical signal shall be classified as dynamic signals with a variable physical length, but limited to a maximum value. Within each signal description the size of data in the signal will be specified but the physical signal length can vary. A dynamic signal must end with an EOS if it is requested in the signal description.

**Dynamic signal indication**

In this specification dynamic signals with a variable physical length are defined with the words "**Byte 1 up to xx**".

**Example (Coding: Table I)**

Specification entry:

**Byte 1 up to 50: *DYNAMIC SIGNAL* name**

Max. 25 characters. 24 letters plus 1 end of string character.

Physical signal layout of dynamic signal:

21 00 30 00 32 00 32 00

22 31 00 39 00 30 00 00 🡸 EOS

**Example (Coding: Table II)**

Specification entry:

**Byte 1 up to 30: *DYNAMIC SIGNAL* name**

Max. 30 characters. 29 letters plus 1 end of string character.

Physical signal layout of dynamic signal:

21 20 30 20 32 20 32 20

22 31 20 39 20 30 20 00 🡸 EOS

### Command Execution Status

The Command Execution Status (CES) codes are used to indicate the status of a response to a request. The usage of each specific CES code shall be defined within each respective channel description. CES codes are separated into four main groups:

|  |  |
| --- | --- |
| **CES Range** | **CES Group** |
| 0x01 – 0x0F | Final Result – Success |
| 0x10 – 0x1F | Final Result – Failure |
| 0x20 – 0x2F | Final Result – Information |
| 0x30 – 0x3F | Intermediate Result – Wait |

#### TP-REQ-015134/A-CES Table (TcSE ROIN-138094-3)

The following table provides a listing of all available CES codes.

|  |  |  |  |
| --- | --- | --- | --- |
| CES | | Parameter Name | Parameter Description |
| High nibble | Low nibble |
| 0 | 0 | INVALID/INACTIVE | Used in request signals with CES field. |
| 0 | 1 | Final Result – Success / update available | General value for this group - Final result Success.  Command processing completed. Final result available. General success for command execution without detailed information. No error occurred.  Example: Requester is requesting list from source. List is present, full and accessible. The information is collected in the source, setup and transmitted within the defined time frame. |
| 0 | 2-F | Reserved | Reserved |
| 1 | 0 | Final Result – Failure | General value for this group - Final result Failure.  Command processing completed. Final result available. General failure for command execution without detailed information. All other failures not specifically listed in this section (CES, High nibble: 1) shall be mapped to this value. |
| 1 | 1 | Final Result – Failure, item missing | Command processing completed. Final result available. Failure for command execution with detailed information.  Fault Information: The requested item is not or no longer available.  Example: The requester is requesting a list from the source. The source is responding with the requested, valid list. After the list is showing at the requester unit, the user deletes a list entry locally at the source. Than the user is requesting a list item that was deleted within the time, but was still visible in the requester list. In this cases the CES parameter: Failure, item missing is to set.  Example:  The track is changed and all track information need to be transmitted, but an ID3 tag (e.g. artist, genre) information is missing. The CES value of not available information is set to this value. |
| 1 | 2 | Final Result – Failure, request released | Command processing completed. Final result available. Failure for command execution with detailed information.  Fault Information: The requested command is not or no longer stored in application. The request is released. The requester could re-init the command execution.  Example:  The command execution is stopped from the receiver unit. No result is presented from the receiver. |
| 1 | 3 | Final Result – Failure, request invalid | Command processing completed. Final result available. Failure for command execution with detailed information.  Fault Information: The requested command (Signal ID) is known, but invalid for the receiver. Parameter combination not possible or unknown.  Example: The requester is transmitting a request to the source. The defined maximum number of characters for a string variable number is 25 characters. The requester is requesting 35 characters from the source. This means the following parameter was not populated with the right values. This parameter combination is invalid. |
| 1 | 4 | Final Result – Failure, requested index out of range | Command processing completed. Final result available. Failure for command execution with detailed information.  Fault Information: The requested index in the received command is out of range.  Example: The protocol allows a maximum of 20 items (0-20). Five bits are reserved for this parameter in the data stream. A range from 0 to 31 is possible.  The requester is asking for 22 items from a list, although 0-20 is only possible. |
| 1 | 5 | Final Result – Failure, connected environment (or device) not reacting | Command processing completed. Final result available. Failure for command execution with detailed information.  Fault Information: The connected environment is not or no longer responding. The command could not be executed.  Example: The requester is asking for a telephone service, but the telephone network from the chosen provider is not available.  Example: The BT connection to the mobile is lost. |
| 1 | 6 | Final Result – Failure, device busy, request released | Command processing completed. Final result available. Failure for command execution with detailed information.  Fault Information: The device or connected device is still busy. Requested application will not attempt to gather the information. The request is released from the application.  Example: The connected device is busy or crashed. Media could not be accessed. |
| 1 | 7 | Reserved | Reserved |
| 1 | 8 | Final Result – Failure, connected device not or no longer readable | Command processing completed. Final result available. Failure for command execution with detailed information.  Fault Information: The connected device is not or no longer readable. The command could not be executed.  Example: The request is transmitted and received. The device reaches a point on the media where it is not readable. A meaningful response could not be given. |
| 1 | 9-F | Reserved | Reserved |
| 2 | 0 | Final Result – Boarder of list reached | Command processing completed. Final result and information related to the command execution is available.  Information: The end or the beginning of the available list is reached.  Example: A list with no wrap around is delivered. If the user tries to brows up although the upper boarder is already reached, the source is responding with this parameter. |
| 2 | 1 | Final Result – List / Folder / Playlist / Selection empty | Command processing completed. Final result and information related to the command execution is available.  Information: The requested list is empty.  Example: The call stack feature is available and supported from the mobile phone. The call stacks from the telephone are requested. The call stacks are empty and have to be reported as empty. |
| 2 | 2 | Final Result – Device empty | Command processing completed. Final result and information related to the command execution is available.  Information: The connected device is empty.  Example: A blank media (USB-MSD, BT-MP, …) is connected. If access and browsing are possible, this response is given. |
| 2 | 3 | Final Result – Feature not supported from node | Command processing completed. Final result and information related to the command execution is available.  Information: The feature connected to the request is not supported by this node or this version of the node.  Example: The BVC is connected to the system. The current software flashed onto the BVC does not support the browsing feature for connected BT media players. If a request arrives to get the BT media player list, this response is given. |
| 2 | 4 | Final Result – Requested command not supported | Command processing completed. Final result and information related to the command execution is available.  Information: The command signal ID is unknown to the receiver.  Example: A request command is received. The TP signal ID is unknown to the receiver. |
| 2 | 5 | Final Result – Status changing | Command processing or status signal update completed. Final result and information related to the command execution is available.  Information: The previous transferred information with this signal ID is invalidated. New information is not yet available. If the new information becomes available, a new signal is transmitted. Within the new signal the CES parameter is set to “Final Result – Success/ Update Available”  Example: The status signal for ActiveFolderName related to the currently played track on the active CD is transmitted. The user request afterwards to change to another CD than the active one. The CD mechanism starts to move. Before the chosen track onto this CD is reached, the ActiveFolderName needs to be invalidated for the system, because during the described changing process, no folder path is available. |
| 2 | 6 | Final Result – Connected environment (or device) not or no longer present | Command processing or status signal update completed. Final result and information related to the command execution is available.  Information: The connected environment is not or no longer present. The command could not be executed.  Example: The request is transmitted and received. The device is un-plugged during command execution.  Example: The requester tries to get information from the phone book what is delivered on the fly from the device. The request is received. Intermediate result is given. Before the final result is reached and all information is collected and transferred, the BT connection is lost or the BT link is switched off. This final result is to transfer. |
| 2 | 7 | Final Result – Feature not supported by connected environment (or device) | Command processing completed. Final result and information related to the command execution is available.  Information: The feature connected to the request is not supported by this device or by this version of the device.  Example: The connected BT phone only supports a blind redial of the last outgoing number without transferring the phone number to the BVC. In this case, if the last outgoing number is requested, this final result is to transfer. |
| 2 | 8 | Final Result – List full; not empty place left in list | Command processing completed. Final result and information related to the command execution is available.  Information: List is full.  Example: User tries to connect a BT device to the system. No empty slot for a BT device is free. This response / status information is given. |
| 2 | 9 | Final Result – No valid Data to proceed | Command processing completed. Final result and information related to the command execution is available.  Information: No valid Data to proceed  Example: The TMC Tuner has lost the TMC Station tuning information. The TMC server shall provide the needed information again. This response / status information is given. |
| 2 | A-F | Reserved | Reserved |
| 3 | 0 | Intermediate Result – Wait | General value for this section - Intermediate result.  Command execution in progress. Final result not yet available. Intermediate result and information related to the command execution is available.  Information: Wait, previous request is executing. The command is known and could be executed. The unit is processing, which means that the application is waiting for unit internal results or is waiting for the connected environment. Intermediate result transferred according to heartbeat strategy.  Example: When scrolling through the Phonebook items, the user may push the Scroll-button faster than the items can be received and displayed. The next Phonebook item is requested not before the previous item is received. This way every single item is displayed to the user. When the user stops pushing the Scroll-button, scrolling stops immediately and no more Phonebook items are displayed. |
| 3 | 1 | Intermediate Result – Wait; device busy, previous received request released; new received request executed. | Command execution in progress. Final result not yet available. Intermediate result and information related to the command execution is available.  Information: Wait, new request is executing. The command is known and could be executed. The unit is processing, which means that the application is waiting for unit internal results or is waiting for the connected environment. This response code indicates that the last received command is executed. All previous commands (with this signal ID) will not be executed. Intermediate result transferred according to heartbeat strategy.  Example: First the user requests to list Phonebook items starting with character 'a', but changes the requested character to 's' before the items are received. Then the request for character 'a' is released and a new request for character 's' is sent. |
| 3 | 2-F | Reserved | Reserved |
| 4-F | 0-F | Reserved | Reserved |

#### TP-REQ-015135/A-CES Reserved Values (TcSE ROIN-138095-1)

If a "Reserved" CES value is received, this value shall be mapped to the "general" value of the related group as defined below:

"Reserved" CES values with a zero in the high nibble are mapped to "Final Result – Success / update available" (CES = 0x01).

"Reserved" CES values with a one in the high nibble are mapped to "Final Result – Failure" (CES = 0x10).

"Reserved" CES values with a two in the high nibble are mapped to "Final Result – Requested Command Not Supported" (CES = 0x24).

"Reserved" CES values with a three in the high nibble are mapped to "Intermediate Result – Wait" (CES = 0x30).

"Reserved" CES values in the 0x40 – 0xFF are mapped to "Invalid" (CES = 0x00).

#### TP-REQ-015136/A-CES Support (TcSE ROIN-138096-1)

All nodes assigned to receiving logical channels containing CES shall implement all values assigned to each channel. An appropriate reaction like re-transmission, show failure, show wait screen or show additional information shall be carried out as defined by CES description and HMI specification.

#### TP-REQ-015137/A-CES Reporting - Final Result – (CES = 0x01-0x2F) (TcSE ROIN-146451-2)

**Responder**

**Success**

If a request can be answered successfully, the high nibble of the parameter "Command execution status" is equal to '0'.

**Failure**

If a request has to be answered as failure, the high nibble of parameter "Command execution status" is equal to '1'.

**Information**

If a request cannot be answered, but additional information to this request is available, the high nibble of parameter "Command execution status" is equal to '2'.

**No Previous Intermediate Response**

When a request is received, and the responder can provide a final result within T\_isoTPrsp, no intermediate result shall be given. No heartbeat function of this signal shall be activated.

**Intermediate Response Finished**

When the final result for a previous sequence of intermediate result ends the heartbeat shall be stopped.

**Signal information**

When the CES parameter is equal to Final Result – Failure (0x10-0x1F) or Final Result - Information (0x20-0x2F) all fields after the CES field will not be transmitted in the response signal. Attached arrays are not filled with information, the array and all attached records are not present, no EOS characters are filled in. Only if the CES parameter is equal to Final Result – Success (0x01), requested data is transmitted.

#### TP-REQ-015138/A-CES Reporting - Intermediate Result (CES = 0x30 – 0x3F) (TcSE ROIN-145779-1)

**Responder**

When a request is received, the responder shall determine if a final result can be provided within T\_isoTPrsp. If a final result can not be provided within this time period then an "Intermediate Result" shall be provided. Once the "Intermediate Result" response sequence is started, the heartbeat function of this signal shall be activated. The heartbeat function is described in the section "Heartbeat".

**Signal Information**

When the CES parameter is equal to Intermediate Result all fields after the CES field shall not be transmitted in the response signal. Attached arrays are not filled with information, the array and all attached records are not present, no EOS characters are filled in, etc.

**Requester**

The requester unit is transmitting a request to the source. The source is answering with an intermediate result. There are two intermediate result responses available and each may have an impact on functions and HMI. Therefore, the following scenarios for the HMI have been defined and the implementation of these scenarios will be addressed at the HMI level. Responses will not swap during runtime, but are fixed to the scenario.

**Response "Intermediate Result – Wait"**

Scenario 1: The requester unit buffers the user input locally and waits for the Response "Final Result" before it sends a new request.

Implementation example: DTMF-Tones are entered in various order by the user, but processing speed is limited by the GSM-network. Then the DTMF-Tones need to be buffered and sent after the Response "Final Result".

Scenario 2: The requester unit increases or decreases the parameter internally until user input for this action ends.

Implementation example: To skip several Tracks, the user rapidly pushes the Skip-button. The number of Tracks to skip is increased in the requester unit until user ends pushing the Skip-button. Then the request is sent out with the actual value of Tracks to skip in total.

Scenario 3: The requester unit does not buffer the user input and waits for the Response "Final Result" before it sends a new request.

Implementation example: When scrolling through the Phonebook items, the user may push the Scroll-button faster than the items can be received and displayed. The next Phonebook item is requested not before the previous item is received. This way every single item is displayed to the user. When the user stops pushing the Scroll-button, scrolling stops immediately and no more Phonebook items are displayed.

**Response " Intermediate Result – Wait; device busy, previous received request released; new received request executed "**

Scenario: The requester unit may send out user input directly with a new request. The requester does not wait for each response to a request. If during the heartbeat period a new request is received, parameters related to the new request shall be updated in the responder. The "Intermediate Result – Wait; device busy, previous received request released; new received request executed" shall be transmitted again to give the requester feedback. The requester shall delete currently received information related to a superseded request, as soon as a new request is placed.

Implementation example 1: The user requests to list Phonebook items starting with character 'a', but changes the requested character to 's' before the items are received. The request for character 'a' is released and a new request for character 's' is sent.

Implementation example 2: The user requests to list Phonebook items starting with character 'a', but changes the requested character to 's' just as the items for 'a' have been received. The request for character 'a' is released and the 'a' information is deleted as soon as the new request for 's' is placed.

**End of Intermediate Result**

The intermediate result sequence is ended if a final result is reached.

#### TP-TMR-REQ-015139/A-T\_isoTPrsp (TcSE ROIN-146458-1)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Units** | **Range** | **Resolution** | **Default** |
| T\_isoTPrsp | Maximum time period allowed for responder to determine if a Final-Result can be provided one a request is received. | msec | 0-100 | 5 | 20 |

## Channel Management

### TP-REQ-015140/A-Concurrent Data Transmission (TcSE ROIN-145774-1)

During an ongoing signal transmission the need for an concurrent data transmission within the same channel could occur. For example, if the user is browsing continuously in a device which is currently playing the channel will be busy transporting the browse data. If, during browsing, the active track is ended and the next track is activated all track information like genre, artist and track name need to be transferred, but the channel is still busy transporting the browse data. The responder shall not interrupt the ongoing data transfer. The responder shall store the pending data until the channel is free. When the channel is free, the pending data shall be transferred in regards to priority and topicality. E.g. if several tracks have been played during browsing, only the track information of the currently active track shall be transferred.

### TP-REQ-015141/A-Multi-Channel Management (TcSE ROIN-199074-1)

A transmitting node may be assigned several physical channels with each connected to different receiving nodes. Also, these physical channels may have the capability to transport the same Logical Signal information to different subscribing nodes (e.g. ArtistName\_St, GetTUPresetInfo\_Rsp). The Logical Signal information is supplied from a server either upon request from a client application (e.g. GetTUPresetInfo.Rq) or upon change of status (i.e. "Pushed") within a server application (CurrentStationName.St).

For information sent ("Pushed") upon status change, the server shall send the data to all subscribing clients. Subscribing clients can be identified through the Signal and Channel Catalog. Within the Catalog, logical signals are assigned to physical channels and physical channels are assigned to transmitting and receiving nodes. Where a logical signal is assigned to more then one physical ID indicates that there are multiple subscribers of this information.

For information sent upon request, the server shall only respond to the specific requesting client and not to all clients capable of issuing the same request.

### Signal Heartbeat

Due to the possibility that it may take an unspecified amount of time for a responder to provide information back to the requester, the "Intermediate Result – Wait" CES code is provided in the response message. To ensure that a requester does not wait forever for a response, a "heartbeat" strategy is employed for all logical signals in the "Wait" state.

#### TP-REQ-015142/A-Signal Heartbeat - Responder (TcSE ROIN-145775-1)

**Responder**

The heartbeat signaling function shall be activated, within the responder, immediately after the first transmission of a signal with an "Intermediate Result" (i.e. CES = 0x30 or 0x31). Once activated, the responder shall, on a periodic basis, re-transmit the "Wait" state signal with an "Intermediate Result". The periodic transmission rate shall be equal to THB\_IntermediateRsp.

If multiple signals are in a "Wait" state, then each signal shall require its own heartbeat signaling function.

When the CES parameter is changed the signal shall be updated and the signal transmitted immediately.

When the CES parameter is equal to "Final Result" either success or failure, the heartbeat signaling function shall be canceled.

#### TP-REQ-015143/A-Signal Heartbeat - Requester (TcSE ROIN-146620-1)

**Requester**

When a requester receives a response with an "Intermediate Result", the requester shall activate a signal heartbeat monitoring session in which the Requester starts a received signal timer. Subsequent receptions of the same Signal ID with an "Intermediate Result" shall reset the respective received signal timer.

If multiple signals are in a "Wait" state, then each signal shall require its own heartbeat monitoring session.

The heartbeat monitoring session shall be canceled when a "Final Result" is received regardless of success or failure.

A signal shall be classified as "missing" if a respective signal is not received after 3 x THB\_IntermediateRsp. Once a signal is determined to be "missing", the heartbeat monitoring session shall be canceled and the original information request released without any retry.

#### TP-REQ-015144/A-Concurrent Data Transmission During Heartbeat Session (TcSE ROIN-146621-1)

During an ongoing heartbeat session other logical signals on the same channel from the responder node to the requester node shall be allowed to occur. The periodic task for the heartbeat message transmission may occur while another signal is currently being transmitted. Since this other signal is originating from the same node, the node can be considered as "Node Fully Operable". Therefore, the reception of another signal allocated to the same channel as the current "wait" state signal shall cause the received signal timer(s) for any signals on the same channel in a monitoring session to be reset.

After the transmission of the other signal, the heartbeat signaling function shall be restarted. If during the transmission of the other signal a final result is reached on the "wait" state signal, the final result shall be transmitted as soon as the bus is free.

The following figure will elaborate on the items previously mentioned.



(1) The request is placed from the requester node onto the bus system. This could be a CAN frame or a ISO-TP message

(2) The responder is answering with CES = Intermediate Result. After the first transmission the heartbeat function is activated. The timer is reloaded with THB\_IntermediateRsp. When the timer expires the next transmission with CES = Intermediate Result is performed. No other TP traffic occurs during the heartbeat transmission.

(3) & (4) After the responder is answering with CES = Intermediate Result other TP traffic occurs on the channel. Due to this other traffic, the heartbeat message transmission must be queued until the channel is free. After the completion of the other TP traffic, the heartbeat signaling function shall be restarted.

#### TP-TMR-REQ-015145/A-THB\_IntermediateRsp (TcSE ROIN-146545-1)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Units** | **Range** | **Resolution** | **Default** |
| THB\_IntermediateRsp | Periodic transmit rate of the heartbeat message for signals in the "wait" state. | msec | 0-1000 | 100 | 1000 |

## Signal and Channel Catalog

### Signal/Channel Mapping Tables

The following section shall provide the signal to channel mapping tables. The mapping tables provide a reference as to which logical signals will be contained in which physical channel. Please note that some logical signals can be contained in several physical channels. Consult the Signal Descriptions section for exact details of each signal.

#### TP-PHY-TPP-REQ-023116/D-AUDIO - RDISP (TcSE ROIN-138122-6)

The AUDIO – RDISP channel represent the signals connecting "AUDIO" features and "RDISP" display features. "AUDIO" represents a Radio and / or CD unit. The "AUDIO" device could be a headunit like IAM/AHU/ACM/ACU. RDISP represents the multimedia display unit. The RDISP device could be an MFD.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: Audio** |  |
| 0x2B4 | AUDIO\_RDISP\_WORD\_Tx | 15 |  |  | **Receiver: RDISP** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x3B | RadioText\_St | AMFM, DAB |
|  |  |  |  | 0x60 | GetStationList\_Rsp | Dynamic Station List |
|  |  |  |  | 0x63 | GetTAGInfo\_Rsp | AMFM |
|  |  |  |  | 0x52 | GetFolderName\_Rsp | CD |
|  |  |  |  | 0x3D | GetMPInfo\_Rsp | CD |
|  |  |  |  | 0x3E | ActiveFolderInfo\_St | CD |
|  |  |  |  | 0x3F | AlbumName\_St | CD |
|  |  |  |  | 0x42 | ArtistName\_St | CD |
|  |  |  |  | 0x43 | FileName\_St | CD |
|  |  |  |  | 0x44 | FolderName\_St | CD |
|  |  |  |  | 0x45 | GenreName\_St | CD |
|  |  |  |  | 0x46 | TrackName\_St | CD |
|  |  |  |  | 0x64 | GetCDTOCData\_Rsp | CD |
|  |  |  |  | 0x6A | EnsembleName\_St | AMFM, DAB |
|  |  |  |  | 0x6B | CurrentStationName\_St | AMFM, DAB |
|  |  |  |  | 0x5F | GetTUPresetInfo\_Rsp | AMFM, SDARS, DAB |
|  |  |  |  | 0x70 | AHU\_Bezel\_Diag\_Data | Data Service2 |
|  |  |  |  | 0x3C | GetPresetInfo2\_Rsp | AMFM, SDARS, DAB |
|  |  |  |  | 0x76 | LBP1\_ItemInfo\_Rsp | AMFM, SDARS, DAB |
|  |  |  |  | 0x8D | RadioText2\_St | DAB |
|  |  |  |  | 0x79 | MediaInformation\_St | AMFM, SDARS, DAB |
|  |  |  |  | 0x92 | DynamicLabelPlus\_St | DAB |
|  |  |  |  | 0x93 | JournalineTxtMsg\_St | DAB |

#### TP-PHY-TPP-REQ-023117/C-SDARS - RDISP (TcSE ROIN-147073-4)

The SDARS – RDISP channel represent the signals connecting "SDARS" features and "RDISP" display features. "SDARS" represents the satellite radio unit. The "SDARS" unit could be integrated into a headunit like IAM/AHU/ACM/ACU. RDISP represents the multimedia display unit. The RDISP device could be an MFD.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: SDARS** |  |
| 0x2C1 | SDARS\_RDISP\_WORD\_Tx | 27 |  |  | **Receiver: RDISP** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x66 | SDARS\_Alert\_St | SDARS |
|  |  |  |  | 0x67 | DispInfo\_ArtistName\_St | SDARS, HD Radio |
|  |  |  |  | 0x68 | SDARS\_CatName\_St | SDARS |
|  |  |  |  | 0x69 | SDARS\_ChannelInfo\_Rsp | SDARS |
|  |  |  |  | 0x6C | SDARS\_ChannelName\_St | SDARS |
|  |  |  |  | 0x6D | SDARS\_CurrentCatList\_Rsp | SDARS |
|  |  |  |  | 0x6E | SDARS\_SetAlert\_Rsp | SDARS |
|  |  |  |  | 0x6F | DispInfo\_SongTitle\_St | SDARS, HD Radio |
|  |  |  |  | 0x73 | SDARS\_ESN\_St | SDARS |
|  |  |  |  | 0x7D | SDARS\_PID\_St | SDARS |
|  |  |  |  | 0x76 | LBP1\_ItemInfo\_Rsp | SDARS |
|  |  |  |  | 0xA2 | SDARS\_ChannelList\_Rq | SDARS |

#### TP-PHY-TPP-REQ-023118/C-RDISP - SDARS (TcSE ROIN-147074-2)

The SDARS – RDISP channel represent the signals connecting "SDARS" features and "RDISP" display features. "SDARS" represents the satellite radio unit. The "SDARS" unit could be integrated into a headunit like IAM/AHU/ACM/ACU. RDISP represents the multimedia display unit. The RDISP device could be an MFD.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: RDISP** |  |
| 0x2C9 | SDARS\_RDISP\_WORD\_Rx | 27 |  |  | **Receiver: SDARS** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x65 | SDARS\_SetAlert\_Rq | SDARS |
|  |  |  |  | 0x80 | ChannelInfo\_Rq | SDARS |
|  |  |  |  | 0x6C | SDARS\_ChannelName\_St | SDARS |
|  |  |  |  | 0xA1 | SDARS\_ChannelList\_Rsp | SDARS |

#### TP-PHY-TPP-REQ-023119/A-SSPServer - SSPClient (TcSE ROIN-147240-1)

The SSPServer - SSPClient channel represent the signals connecting "Sirius Data Services" client and the server. This channel is used by the server to transport data to the client. The SSPClient may exist in a display module (e.g. MFD) and the SSPServer may exist in the AHU/ACM.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: AHU** |  |
| 0x2C3 | SSPClnt\_SSPSrv\_WORD\_Tx | 23 |  |  | **Receiver: APIM** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x62 | SSP\_Rsp | SDARS |

#### TP-PHY-TPP-REQ-023120/A-SSPClient - SSPServer (TcSE ROIN-147262-1)

The SSPClient – SSPServer channel represent the signals connecting "Sirius Data Services" client and the server. This channel is used by the client to transport data to the server. The SSPClient may exist in a display module (e.g. MFD) and the SSPServer may exist in the AHU/ACM.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: APIM** |  |
| 0x2CB | SSPClnt\_SSPSrv\_WORD\_Rx | 23 |  |  | **Receiver: AHU** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x61 | SSP\_Rq | SDARS |

#### TP-PHY-TPP-REQ-023121/A-Bezel - RDISP (TcSE ROIN-147429-1)

The Bezel – RDISP channel represent the signals connecting Bezel features and "RDISP" display features. Bezel represents a button input panel. RDISP represents the multimedia display unit. The RDISP device could be an MFD.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: FCIM** |  |
| 0x2C6 | Bezel\_RDISP\_WORD\_TX | 26 |  |  | **Receiver: MFD** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x71 | EFP\_Bezel\_Diag\_Data | Data\_Services2 |

#### TP-PHY-TPP-REQ-015146/A-DSPAMP - RDISP (TcSE ROIN-147559-1)

The DSPAMP – RDISP channel represent the signals connecting DSPAMP features and "RDISP" display features. DSPAMP represents a remote audio amplifier. RDISP represents the multimedia display unit. The RDISP device could be an MFD.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: DSP** |  |
| 0x2C5 | DSPAMP\_RDISP\_WORD\_TX | 25 |  |  | **Receiver: MFD** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x72 | DSP\_Bezel\_Diag\_Data | Data\_Services2 |

#### TP-PHY-TPP-REQ-023122/A-RDISP - AUDIO (TcSE ROIN-147574-1)

The RDISP – AUDIO channel is representing the channel connecting "RDISP" features and "AUDIO" features. The "RDISP" represents the multimedia display unit. The RDISP device could be a MFD or a headunit. "AUDIO" represents a Radio and / or CD unit. The "AUDIO" device could be a headunit like IAM/AHU/ACM/ACU.

This channel is used for Flow Control only. Requests from RDISP to Audio are transferred in Single-CAN-Frames.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: MFD** |  |
| 0x2BC | AUDIO\_RDISP\_WORD\_RX | 15 |  |  | **Receiver: AUDIO** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | -- | -- | -- |

#### TP-PHY-TPP-REQ-013860/B-TMCServer - TMCClient (TcSE ROIN-159708-3)

The TMCServer – TMCClient channel is representing the channel connecting "TMCServer" features and "TMCClient" features. The "TMCClient" represents the multimedia display unit. The TMCClient device could be a MFD or a headunit. "TMCServer" represents a TMC Tuner unit. The "TMCServer" device could be a headunit like IAM/AHU/ACM/ACU.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: AHU** |  |
| 0x2C4 | TMCServer\_TMCClient\_WORD\_Tx | 33 |  |  | **Receiver: MFD** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x74 | TMCData\_St | TMC Data |
|  |  |  |  | 0x7A | TMCServiceProvider\_St | TMC Data |
|  |  |  |  | 0x99 | TrafficServicProvider\_St | TMC Data |

#### TP-PHY-TPP-REQ-023123/A-LBP1Server - LBPClient (DELETED) (TcSE ROIN-159926-2)

#### TP-PHY-TPP-REQ-023124/C-NavRepServer - NavRepClient (TcSE ROIN-160780-1)

The NavRepServer – NavRepClient channel is representing the channel connecting "Navigation Repeater Server" features and "Navigation Repeater Client" features. The "Navigation Repeater Server" represents a navigation unit. "Navigation Repeater Client" represents a display module.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: See CAN database** |  |
| 0x2C0 | NAV\_MC\_WORD\_Tx | 31 |  |  | **Receiver: See CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x77 | Destination\_Info\_St | Navigation |
|  |  |  |  | 0x78 | CurrentStreetName\_St | Navigation |
|  |  |  |  | 0x22 | NavigationSymbolInfo\_St | Navigation |
|  |  |  |  | 0x20 | StreetName\_St | Navigation |

#### TP-PHY-TPP-REQ-023125/C-MediaPlayerServer - MediaPlayerClient (TcSE ROIN-160781-2)

The MediaPlayerServer – MediaPlayerClient channel is representing the channel connecting "Media Player Server" features and "Media Player Client" features. The "Media Player Server" represents a media playback unit. "Media Player Client" represents an instrument cluster.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: See CAN database** |  |
| 0x2B7 | CONMP\_MC\_WORD\_Tx | 18 |  |  | **Receiver: See CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x79 | MediaInformation\_St | Generic Metadata |
|  |  |  |  | 0x76 | LBP1\_ItemInfo\_Rsp | List Browser Data |
|  |  |  |  | 0x5F | GetTUPresetInfo\_Rsp | AMFM, SDARS, DAB |
|  |  |  |  | 0x67 | DispInfo\_ArtistName\_St | SDARS |
|  |  |  |  | 0x68 | SDARS\_CatName\_St | SDARS |
|  |  |  |  | 0x6C | SDARS\_ChannelName\_St | SDARS |
|  |  |  |  | 0x6F | DispInfo\_SongTitle\_St | SDARS |

#### TP-PHY-TPP-REQ-023126/C-PHONE - MC (TcSE ROIN-160782-3)

The PHONE – MC channel is representing the channel connecting "PHONE" features and "MC" features. The "MC" represents a multimedia display unit. "PHONE" represents a phone interface module.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: See CAN database** |  |
| 0x2B6 | PHONE\_MC\_WORD\_Tx | 17 |  |  | **Receiver: See CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x50 | BTCallerIdentification\_St | Phone |
|  |  |  |  | 0x4F | InitiateBTCall\_Rsp | Phone |

#### TP-PHY-TPP-REQ-023127/A-MC - PHONE (TcSE ROIN-160783-2)

The MC – PHONE channel is representing the channel connecting "MC" features and "PHONE" features. The "MC" represents a multimedia display unit. "PHONE" represents a phone interface module.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: IPC** |  |
| 0x2BE | PHONE\_MC\_WORD\_Rx | 17 |  |  | **Receiver: MFD** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x0D | InitiateBTCall\_Rq | Phone |
|  |  |  |  |  |  |  |

#### TP-PHY-TPP-REQ-023128/B-TMCClient - TMCServer (TcSE ROIN-178758-2)

The TMCClient – TMCServer channel is representing the channel connecting "TMCClient" features and "TMCServer" features. The "TMCClient" represents the multimedia display unit. The TMCClient device could be a MFD or a headunit. "TMCServer" represents a TMC Tuner unit. The "TMCServer" device could be a headunit like IAM/AHU/ACM/ACU.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: MFD** |  |
| 0x2CC | TMCServer\_TMCClient\_WORD\_Rx | 33 |  |  | **Receiver: AHU** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x7B | TMCGetServiceProvider\_Rq | TMC Data |
|  |  |  |  | 0x9A | TrafficeGetServiceProvider\_Rq | TMC Data |

#### TP-PHY-TPP-REQ-023129/A-EnvStatReportServer - EnvStatReportClient (TcSE ROIN-201966-1)

The EnvStatReportServer – EnvStatReportClient channel represent the signals connecting "EnvStatReportServer " features and "EnvStatReportClient " features. "EnvStatReportServer " represents a Data Report unit. The "EnvStatReportServer " Could be a part of the IPC. "EnvStatReportClient" could be a display unit.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: see CAN database** |  |
| 0x2C7 | RepSrv\_RepClient\_WORD\_Tx | 24 |  |  | **Receiver: see CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Data Field ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x7C | MyKeyReportCardOutput\_Rsp | DataReport |
|  |  |  |  |  |  |  |

#### TP-PHY-TPP-REQ-023130/A-EnvStatReportClient - EnvStatReportServer (TcSE ROIN-201967-1)

The EnvStatReportClient - EnvStatReportServer channel represent the signals connecting "EnvStatReportServer " features and "EnvStatReportClient " features. "EnvStatReportServer " represents a Data Report unit. The "EnvStatReportServer " Could be a part of the IPC. "EnvStatReportClient" could be a display unit.

This channel is for Flow Control only.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: see CAN database** |  |
| 0x2CF | RepSrv\_RepClient\_WORD\_Rx | 24 |  |  | **Receiver: see CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Data Field ID** | **Signal Name** | **Utilization** |
|  |  |  |  | -- | -- | -- |

#### TP-PHY-TPP-REQ-023131/C-APIM - TCU (TcSE ROIN-223472-2)

The APIM – TCU channel is representing the channel connecting "TCU" features and "APIM" features.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: APIM** |  |
| 0x4A8 | APIM\_TCU\_Word\_Tx | 20 |  |  | **Receiver: TCU** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x82 | ChargeProfileList\_Rq | Charge Programming |
|  |  |  |  | 0x84 | ChargeSchedule\_Rq | Charge Programming |
|  |  |  |  | 0x86 | SyncSoftwareVersion\_Rsp | Charge Programming |
|  |  |  |  | 0x94 | WifiInfo\_Rq | Embedded Modem |
|  |  |  |  | 0xA3 | MapVersionNumber\_St | Nav\_Service2 - Navigation |
|  |  |  |  |  |  |  |

#### TP-PHY-TPP-REQ-023132/D-TCU - APIM (TcSE ROIN-223473-3)

The TCU – APIM channel is representing the channel connecting "TCU" features and "APIM" features.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: TCU** |  |
| 0x4A0 | APIM\_TCU\_Word \_Rx | 20 |  |  | **Receiver: APIM** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x81 | CabinComfortPreferenceList\_Rsp | Charge Programming |
|  |  |  |  | 0x83 | ChargeProfileList\_Rsp | Charge Programming |
|  |  |  |  | 0x85 | ChargeSchedule\_Rsp | Charge Programming |
|  |  |  |  | 0x87 | TelServESN\_St | Charge Programming |
|  |  |  |  | 0x88 | TelServUserID\_St | Charge Programming |
|  |  |  |  | 0x90 | EmergencyCallText\_St | Embedded Modem |
|  |  |  |  | 0x95 | WifiInfo\_Rsp | Embedded Modem |
|  |  |  |  | 0x96 | CarrierInfo\_Rsp | Embedded Modem |
|  |  |  |  | 0x97 | DataUsage\_Rsp | Embedded Modem |
|  |  |  |  | 0x98 | DeviceList\_Rsp | Embedded Modem |
|  |  |  |  | 0x9B | WifiHotspotMAC\_Rsp | Embedded Modem |
|  |  |  |  |  |  |  |

#### TP-PHY-TPP-REQ-023133/A-MC - RDISP (TcSE ROIN-205503-2)

The MC – RDISP channel represent the signals connecting "RDISP" features and "MC" display features. "RDISP" represents a display unit. The "RDISP" device could be a MFD or a headunit. MC represents the Message Center unit. The MC device could be part of the instrument cluster.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: see CAN database** |  |
| 0x2A4 | MC\_RDISP\_WORD\_Tx | 29 |  |  | **Receiver: see CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x7E | MCEventUpdate\_St | -- |
|  |  |  |  | 0x7F | MCGetData\_Rsp | -- |
|  |  |  |  | 0x89 | ConsHistGraph\_St | Electrification Information |

#### TP-PHY-TPP-REQ-023134/A-RDISP - MC (TcSE ROIN-205504-1)

The RDISP – MC channel is representing the channel connecting "RDISP" features and "MC" features. The "RDISP" represents the multimedia display unit. The RDISP device could be a MFD or a headunit. MC represents the Message Center unit. The MC device could be part of the instrument cluster.

This channel is used for Flow Control only. Requests from RDISP to MC are transferred in Single-CAN-Frames.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: see CAN database** |  |
| 0x2AC | MC\_RDISP\_WORD\_Rx | 29 |  |  | **Receiver: see CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | -- | -- | -- |

#### TP-PHY-TPP-REQ-023135/B-CD - RDISP (TcSE ROIN-206152-1)

The CD – RDISP channel represent the signals connecting "AUDIO" features and "RDISP" display features. "CD" represents the Remote CD unit. RDISP represents the multimedia display unit. The RDISP device could be an MFD.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: CD** |  |
| 0x2A6 | CD\_RDISP\_WORD\_Tx | 34 |  |  | **Receiver: RDISP** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x52 | GetFolderName\_Rsp | CD |
|  |  |  |  | 0x3D | GetMPInfo\_Rsp | CD |
|  |  |  |  | 0x3E | ActiveFolderInfo\_St | CD |
|  |  |  |  | 0x3F | AlbumName\_St | CD |
|  |  |  |  | 0x42 | ArtistName\_St | CD |
|  |  |  |  | 0x43 | FileName\_St | CD |
|  |  |  |  | 0x44 | FolderName\_St | CD |
|  |  |  |  | 0x45 | GenreName\_St | CD |
|  |  |  |  | 0x46 | TrackName\_St | CD |
|  |  |  |  | 0x64 | GetCDTOCData\_Rsp | CD |
|  |  |  |  | 0x76 | LBP1\_ItemInfo\_Rsp | CD |
|  |  |  |  | 0x79 | MediaInformation\_St | CD |

#### TP-PHY-TPP-REQ-023136/A-RDISP - CD (TcSE ROIN-206153-1)

The CD – RDISP channel is representing the channel connecting "RDISP" features and "CD" features. The "RDISP" represents the multimedia display unit. The RDISP device could be a MFD or a headunit. "CD" represents the Remote CD unit.

This channel is used for Flow Control only. Requests from RDISP to CD are transferred in Single-CAN-Frames.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: MFD** |  |
| 0x2AE | CD\_RDISP\_WORD\_RX | 34 |  |  | **Receiver: CD** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | -- | -- | -- |

#### TP-PHY-TPP-REQ-092284/A-NavRepServer - NavRepClient2

The NavRepServer – NavRepClient2 channel is representing the channel connecting "Navigation Repeater Server" features and "Navigation Repeater Client2" features. The "Navigation Repeater Server" represents a navigation unit. "Navigation Repeater Client2" represents a display module.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: See CAN database** |  |
| 0x241 | NAV\_RDISP2\_WORD\_Tx | 42 |  |  | **Receiver: See CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x77 | Destination\_Info\_St | Navigation |
|  |  |  |  | 0x78 | CurrentStreetName\_St | Navigation |
|  |  |  |  | 0x22 | NavigationSymbolInfo\_St | Navigation |
|  |  |  |  | 0x20 | StreetName\_St | Navigation |
|  |  |  |  | 0x91 | UpcomingStreetName\_St | Electronic Horizon |

#### TP-PHY-TPP-REQ-092285/A-NavRepClient2 - NavRepServer

The NavRepClient2 - NavRepServer channel is representing the channel connecting "Navigation Repeater Server" features and "Navigation Repeater Client2" features. The "Navigation Repeater Server" represents a navigation unit. "Navigation Repeater Client2" represents a display module.

This channel is used for Flow control only

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: See CAN database** |  |
| 0x249 | NAV\_RDISP2\_WORD\_Rx | 42 |  |  | **Receiver: See CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | -- | -- | -- |

#### TP-PHY-TPP-REQ-092286/B-MediaPlayerServer - MediaPlayerClient2

The MediaPlayerServer – MediaPlayerClient2 channel is representing the channel connecting "Media Player Server" features and "Media Player Client2" features. The "Media Player Server" represents a media playback unit. "Media Player Client2" represents a Heads Up Display.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: See CAN database** |  |
| 0x242 | CONMP\_RDISP2\_WORD\_Tx | 43 |  |  | **Receiver: See CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x79 | MediaInformation\_St | Generic Metadata |
|  |  |  |  | 0x67 | DispInfo\_ArtistName\_St | SDARS |
|  |  |  |  | 0x6C | SDARS\_ChannelName\_St | SDARS |
|  |  |  |  | 0x6F | DispInfo\_SongTitle\_St | SDARS |

#### TP-PHY-TPP-REQ-092287/A-MediaPlayerClient2 - MediaPlayerServer

The MediaPlayerClient2 - MediaPlayerServer channel is representing the channel connecting "Media Player Server" features and "Media Player Client2" features. The "Media Player Server" represents a media playback unit. "Media Player Client2" represents a Heads Up Display.

This channel is used for Flow control only

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: See CAN database** |  |
| 0x24A | CONMP\_RDISP2\_WORD\_Rx | 43 |  |  | **Receiver: See CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | -- | -- | -- |

#### TP-PHY-TPP-REQ-092288/A-PHONE - RDISP2

The PHONE – RDISP2 channel is representing the channel connecting "PHONE" features and "RDISP2 " features. The "RDISP2 " represents a Heads Up display unit. "PHONE" represents a phone interface module.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: See CAN database** |  |
| 0x243 | PHONE\_RDISP2\_WORD\_Tx | 44 |  |  | **Receiver: See CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x50 | BTCallerIdentification\_St | Phone |

#### TP-PHY-TPP-REQ-092289/A-RDISP2 - PHONE

The RDISP2 – PHONE channel is representing the channel connecting "RDISP2 " features and "PHONE" features. The "RDISP2 " represents a Heads Up display unit. "PHONE" represents a phone interface module.

This channel is used for Flow control only

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: AHUD** |  |
| 0x24B | PHONE\_RDISP2\_WORD\_Rx | 44 |  |  | **Receiver: APIM** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | -- | -- | -- |

#### TP-PHY-TPP-REQ-092294/B-MediaPlayerServer - MediaPlayerClient3

The MediaPlayerServer – MediaPlayerClient3 channel is representing the channel connecting "Media Player Server" features and "Media Player Client3" features. The "Media Player Server" represents a media playback unit. "Media Player Client3" represents a Rear display unit.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: See CAN database** |  |
| 0x256 | CONMP\_RSE\_WORD\_Tx | 47 |  |  | **Receiver: See CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | 0x79 | MediaInformation\_St | Generic Metadata |
|  |  |  |  | 0x67 | DispInfo\_ArtistName\_St | SDARS |
|  |  |  |  | 0x68 | SDARS\_CatName\_St | SDARS |
|  |  |  |  | 0x6C | SDARS\_ChannelName\_St | SDARS |
|  |  |  |  | 0x6F | DispInfo\_SongTitle\_St | SDARS |

#### TP-PHY-TPP-REQ-092295/A-MediaPlayerClient3 - MediaPlayerServer

The MediaPlayerClient3 - MediaPlayerServer channel is representing the channel connecting "Media Player Server" features and "Media Player Client3" features. The "Media Player Server" represents a media playback unit. "Media Player Client3" represents a Rear display unit.

This channel is used for Flow control only

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Channel** | | | | | | |
| **CAN ID** | **Msg Name** | **TP Index** |  |  | **Transmitter: See CAN database** |  |
| 0x25E | CONMP\_RSE\_WORD\_Rx | 47 |  |  | **Receiver: See CAN database** |  |
|  |  |  |  | **Logical Signals** | | |
|  |  |  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  |  |  | -- | -- | -- |

#### TP-PHY-TPP-REQ-207117/A-TRAFFIC - RDISP

The TRAFFIC – RDISP channel represent the signals connecting "TRAFFIC" features and "RDISP " display features. "TRAFFIC" represents the traffic logic typical deployed to an embedded modem like the TCU. The "RDISP" device could be a center stack unit (with navigation) like APIM

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Channel** | | | | |
| **Msg Name** |  |  | **Transmitter: TRAFFIC** |  |
| TRAFFIC\_RDISP\_WORD\_Tx |  |  | **Receiver: RDISP** |  |
|  |  | **Logical Signals** | | |
|  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  | 0xCF | megaTP\_ConsecutivePackage | MobileCom\_Service2 - Embedded Modem |
|  |  | 0xFF | megaTP\_FirstPackage | MobileCom\_Service2 - Embedded Modem |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

#### TP-PHY-TPP-REQ-207118/A-RDISP - TRAFFIC

The TRAFFIC – RDISP channel represent the signals connecting "TRAFFIC" features and "RDISP " display features. "TRAFFIC" represents the traffic logic typical deployed to an embedded modem like the TCU. The "RDISP" device could be a center stack unit (with navigation) like APIM

This channel is used for Flow control only

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Channel** | | | | |
| **Msg Name** |  |  | **Transmitter: RDISP** |  |
| TRAFFIC\_RDISP\_WORD\_Rx |  |  | **Receiver: Traffic** |  |
|  |  | **Logical Signals** | | |
|  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  | -- | -- | -- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

#### TP-PHY-TPP-REQ-207115/A-OPTIN - RDISP

The OPTIN – RDISP channel represent the signals connecting "OPTIN" features and "RDISP " display features. "OPTIN" represents the customer opt-in logic typical deployed to an embedded modem like the TCU. The "RDISP" device could be a center stack unit like APIM or CHR.

This channel is used for Flow control

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Channel** | | | | |
| **Msg Name** |  |  | **Transmitter: Customer OPT-IN (OPTIN)** |  |
| OPTIN\_RDISP\_WORD\_Tx |  |  | **Receiver: RDISP** |  |
|  |  | **Logical Signals** | | |
|  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  | 0x9D | CCOISynchronizationSettings\_Rsp | MobileCom\_Service2 - Embedded Modem |
|  |  | 0x9E | CCOISynchronizationAuthorizedUsers\_Rsp | MobileCom\_Service2 - Embedded Modem |
|  |  | 0xA5 | CCOIUserPrompt\_Rq | MobileCom\_Service2 - Embedded Modem |
|  |  | 0xCF | megaTP\_ConsecutivePackage | MobileCom\_Service2 - Embedded Modem |
|  |  | 0xFF | megaTP\_FirstPackage | MobileCom\_Service2 - Embedded Modem |
|  |  |  |  |  |

#### TP-PHY-TPP-REQ-207116/A-RDISP - OPTIN

The OPTIN – RDISP channel represent the signals connecting "OPTIN" features and "RDISP " display features. "OPTIN" represents the customer opt-in logic typical deployed to an embedded modem like the TCU. The "RDISP" device could be a center stack unit like APIM or CHR.

This channel is used for Flow control

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Channel** | | | | |
| **Msg Name** |  |  | **Transmitter: RDISP** |  |
| OPTIN\_RDISP\_WORD\_Rx |  |  | **Receiver: Customer OPT-IN (OPTIN)** |  |
|  |  | **Logical Signals** | | |
|  |  | **Signal ID** | **Signal Name** | **Utilization** |
|  |  | 0x9C | CCOIOnBoardSynchronizationSession\_Rq | MobileCom\_Service2 - Embedded Modem |
|  |  | 0x9F | CCOISynchronizationSummaryReport | MobileCom\_Service2 - Embedded Modem |
|  |  | 0xA0 | CCOISettingsUpdate\_Rq | MobileCom\_Service2 - Embedded Modem |
|  |  | 0xA6 | CCOIUserPrompt\_Rsp | MobileCom\_Service2 - Embedded Modem |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

### Signal Descriptions

#### TP-LOG-TPL-REQ-023137/A-SID-3D-GetMPInfo\_Rsp (TcSE ROIN-138040-2)

Data size: up to 949/549 (Coding Table I / Coding Table II) bytes.

**Byte 0: Signal identifier**

0x3D: GetMPListItemInformation\_Rsp

**Byte 1: Utilization**

0x11: MP\_Media1 – CD

0x12: MP\_Media2 – BT Audio Streaming

0x13: MP\_Media3 – USB

0x15: MP\_Media5 – SD

0x16: MP\_Media6 – DVD

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4-5: FolderNumber**

0x0000 – Root  
0x0001 – Folder number 1  
0x0002 – Folder number 2  
…  
0xFFFF – Folder number 65535

**Byte 6-7: HeaderInfo\_***ItemsInFolder*

*Bit 0-15: ItemsInFolder*

Value: 0..65535

**Byte 8: HeaderInfo\_***NumberOfItems*

*Bit 0-2: reserved*

*Bit 3-7: NumberOfItems*

Value: 0..20

**Byte 9 up to 948/548 (Coding Table I / Coding Table II): Item Info**

Array(1..NumberOfItems) of record (ItemIndex, ItemName, ItemSort, ItemNumber)

*NOTE:  
Transfer starts at ItemIndex sent in the request.  
NumberOfItems value is connected to the value stated in the request.  
If fewer items are available then requested, the NumberOfItems parameter in the response is set to the number of items which are available.*

Record definition (47/27 (Coding Table I / Coding Table II) bytes):

*Bit 0-15: ItemIndex*

Value: 0..65535

*Bit 16 up to 335/175: ItemName*

Max. 20 characters, 19 letters plus 1 end of string character.

*Bit 32/24..336/176 up to 39/31..343/183: ItemSort*

0x0 – Folder

0x1 – File

0x2 – Playlist

0x3 – Videofile

0x4 – Imagefile

0x5 – reserved

…

0x7 - reserved

*Bit 40/32..344/184 up to 71/63..375/215: ItemNumber*

*FolderNumber*

0x0000 – Root  
0x0001 – Folder number 1  
0x0002 – Folder number 2  
…  
0xFFFF – Folder number 65535

*TrackNumber*

*NOTE:  
If ItemNumber is equal to Folder or Playlist the parameter TrackNumber is set to 0x0000. If ItemNumber is fordna3*

*equal to File, the parameter TrackNumber reflects the track number of the selected folder.*

0x0001 – Track 1  
0x0002 – Track 2  
…  
0xFFFF – Track 65535

#### TP-LOG-TPL-REQ-023138/A-SID-6A-EnsembleName\_St (TcSE ROIN-147415-3)

Data size: up to 38/21 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x6A: EnsembleName\_St

**Byte 1: Utilization**

0x01: Radio\_Service1 – AmFm Radio General

0x03: Radio\_Service3 – DAB

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char) - RDS Latin shall be used.

**Byte 4 up to 37/20 (Coding Table I / Coding Table II): Active Ensemble Name**

Max. 17 characters, 16 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023139/A-SID-6B-CurrentStationName\_St (TcSE ROIN-147417-4)

Data size: up to 38/21 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x6B: CurrentStationName\_St

**Byte 1: Utilization**

0x01: Radio\_Service1 – AmFm Radio General

0x03: Radio\_Service3 – DAB

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char) - RDS Latin shall be used.

**Byte 4 up to 37/20 (Coding Table I / Coding Table II): current station name**

Max. 17 characters, 16 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023140/A-SID-42-ArtistName\_St (TcSE ROIN-138041-2)

Data size: up to 44/24 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x42: NameOfArtist\_St

**Byte 1: Utilization**

0x11: MP\_Media1 – CD

0x12: MP\_Media2 – BT Audio Streaming

0x13: MP\_Media3 – USB

0x14: MP\_Media4 – iPod

0x15: MP\_Media5 – SD

0x16: MP\_Media6 – DVD

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 43/23 (Coding Table I / Coding Table II): Active artist name**

Max. 20 characters, 19 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023141/A-SID-43-FileName\_St (TcSE ROIN-146556-2)

Data size: up to 44/24 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x43: NameOfFile\_St

**Byte 1: Utilization**

0x11: MP\_Media1 – CD

0x12: MP\_Media2 – BT Audio Streaming

0x13: MP\_Media3 – USB

0x15: MP\_Media5 – SD

0x16: MP\_Media6 – DVD

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 43/23 (Coding Table I / Coding Table II): Current file name**

Max. 20 characters, 19 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023142/A-SID-3F-AlbumName\_St (TcSE ROIN-146555-2)

Data size: up to 44/24 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x3F: NameOfAlbum\_St

**Byte 1: Utilization**

0x11: MP\_Media1 – CD

0x12: MP\_Media2 – BT Audio Streaming

0x13: MP\_Media3 – USB

0x14: MP\_Media4 – iPod

0x15: MP\_Media5 – SD

0x16: MP\_Media6 – DVD

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 43/23 (Coding Table I / Coding Table II): Current album name**

Max. 20 characters, 19 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023143/A-SID-3E-ActiveFolderInfo\_St (TcSE ROIN-138042-2)

Data size: up to 44 bytes.

**Byte 0: Signal identifier**

0x3E: ActiveFolderInfo\_St

**Byte 1: Utilization**

0x11: MP\_Media1 – CD

0x12: MP\_Media2 – BT Audio Streaming

0x13: MP\_Media3 – USB

0x15: MP\_Media5 – SD

0x16: MP\_Media6 – DVD

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3-11: Item Info**

Array(NbrOfTypes) of record (Type, NbrOfItems)

*Bit 0-4: reserved*

*Bit 5 - 7: Type*

0x0 – Folder

0x1 – File

0x2 – Playlist

0x3-0x7 – *reserved*

*Bit 8-23: NbrOfItem*

0x0000 – No item

0x0001 – Item 1

0x0002 – Item 2

…

0xFFFF – Item 65535

**Byte 12 up to 43: Folder Path (depending onto Folder depth)**

Array(ActiveFolder, Root, … way to ActiveFolder) of record (FolderNumber, ItemIndex)

*Bit 0-15: FolderNumber*

0x0000 – Root

0x0001 – Folder number 1

0x0002 – Folder number 2

…

0xFFFF – Folder number 65535

*Bit 16-31: ItemIndex*

0x0000 – Item position 0 in folder

0x0001 – Item position 1 in folder

0x0002 – Item position 2 in folder

…

0xFFFF – Item position 65535 in folder

#### TP-LOG-TPL-REQ-023144/A-SID-5F-GetTUPresetInfo\_Rsp (TcSE ROIN-146501-5)

Data size: up to 1116/606 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x5F: GetTUPresetInfo\_Rsp

**Byte 1: Utilization**

0x01: Radio\_Service1 – AmFm Radio General

0x02: Radio\_Service2 – SDARS

0x03: Radio\_Service3 – DAB

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char) - RDS Latin shall be used.

**Byte 4-5: Header info**

*Bit 0-7: ListSize*

0x00 – Invalid

0x01 – List Size 1  
0x02 – List Size 2  
…

0x1E – List Size 30

0xFF – No entry

*NOTE: ListSize maximum is currently limited to 30 items.*

*Bit 8-11: reserved*

*Bit 12-15: Preset bank*

0x00 – Not Valid

0x01 – Preset Bank I – FM1

0x02 – Preset Bank II – FM2

0x03 – Preset Bank III – FM3

0x04 – Preset Bank IV – FM AST

0x05 – Preset Bank V – AM

0x06 – Preset Bank VI – AM AST

~~0x07 – Reserved – Reserved~~

0x07 – Preset Bank XII – DAB3

0x08 – Preset Bank VII – DAB1

0x09 – Preset Bank VIII – DAB2

0x0A – Preset Bank IX – SAT1

0x0B – Preset Bank X – SAT2

0x0C – Preset Bank XI – SAT3

**Byte 6 up to 1115/605 (Coding Table I / Coding Table II): Preset Info**

*NOTE: Transfer starts at Preset number sent in the request****N*** *is also stated in the request. If* ***N*** *is greater than ListSize, the complete list will be transferred.*

Array (1...N) of record (PresetNumber, Frequency, Station Name,)

Record definition (37/20 (Coding Table I / Coding Table II) bytes):

*Byte 0: PresetNumber*

0x00 – Reserved  
0x01 – Preset 1  
0x02 – Preset 2  
...

0x1E – Preset 30

0xFF – No valid preset

*Byte 1-2: Frequency*

0x0000 – 0

0x0001 – 1

…

0x0615 – 1557  
0x0616 – Reserved

…

0xFFFF – Reserved

*AM: Freq = 153+ Offset kHz. Offset 0..1557*

*FM: Freq = 76+ Offset\*0.05 MHz. Offset 0..640*

*Selected tuned band determine frequency (kHz or MHz).*

*SDARS: ChanNum = xxx (range = 000 – 223)*

*DAB: frequency/BlockNumber = Bitfield:*

*Bit 0 .. 4:*

*L-Band Canada: Numeric value(1 .. 23);*

*L-Band Europe: Numeric value(A=1 .. W=23);*

*Band III: Numeric value(A=1 .. W=23); hex coded*

*Bit 5 .. 8:*

*Band III: Numeric value (not used for L-Band; default value: 0h), hex coded*

*Bit 9:*

*0: Band III, 1: L-Band*

*Byte 3 up to 36/19 (Coding Table II / Coding Table I): Station Name*

Up to 17 characters 16 letters plus 1 End Of String character

*NOTE: If the Station Name is shorter than 16 characters, it must be terminated with an End Of String.*

*PSName = 8 Characters Max*

*DAB Service Name = 16 Characters Max*

*HD Station Name = 16 Characters Max*

*SDARS Channel name = Short name*

*Short Name = 8 Characters Max*

#### TP-LOG-TPL-REQ-023145/A-SID-60-GetStationList\_Rsp (TcSE ROIN-146502-6)

Data size: up to 1178/668 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x60: GetStationList\_Rsp

**Byte 1: Utilization**

0x04: Radio\_Service4 – Dynamic Station List

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char) - RDS Latin shall be used.

**Byte 4-7: Header info**

*Bit 0: Refresh flag*

$0: False

$1: True

*NOTE: Refresh Flag shall always be set to 0x1 for Single Tuners.*

*Bit 1-4: reserved*

*Bit 5 - 7: StationList*

0x0 – Invalid

0x1 – Analog AM List

0x2 – Analog FM List

0x3 – FM HD List

0x4 – Analog and HD FM List

0x5 – Analog FM PTY List

0x6 – DAB Ensemble Service List

0x7 – DAB Service List

*Bit 8-15: ListSize*

0x00 – Invalid

0x01 – Item 1  
0x02 – Item 2  
…

0xFD – Item 253

0xFE – Not Used

0xFF – No entry

*NOTE: The parameter List Size defines how many list items are transmitted in the response.*

*ListSize maximum is currently limited to 30 items.*

*Bit 16-23: TotalNumOfStatAval*

0x00 – Invalid

0x01 – Item 1  
0x02 – Item 2  
…

0xFD – Item 253

0xFE –Not Used

0xFF – No entry

*Bit 24* – *25: reserved*

*Bit 26 - 31: Requested PTY*

0x00 – Invalid

0x01 – PTY Code  
0x02 – PTY Code  
…

0x1F – PTY Code

*NOTE: If parameter StationList = 0x5-0x7 or 0x1 then Requested PTY = 0x0, else PTY = 0x00 ... 0x1F.**For parameter StationList = 0x2 with FM RDS the Requested PTY = 0x0.*

**Byte 8 up to 1177/667 (Coding Table I / Coding Table II): ItemVector**

*NOTE: Transfer starts at Index Number sent in the request.****N*** *is also stated in the request. If* ***N*** *is greater than ListSize, the complete list will be transferred.*

Array (1...N) of record (IndexNumber, Frequency, HDMulticast, TP Status, TMC Status, Program Type, Station Name)

Record definition (39/22 (Coding Table I / Coding Table II) bytes):

*Bit 0-7: IndexNumber*

0x00 – Reserved  
0x01 – Index 1  
0x02 – Index 2  
...

0xFE – Ensemble Name

0xFF – Invalid

*Bit 8-23: Frequency*

0x000 – 0

0x001 – 1

…

0x615 – 1557  
0x616 – reserved

…

0xFFFF – reserved

*AM: Freq = 153+ Offset kHz. Offset 0..1557*

*FM: Freq = 76+ Offset\*0.05 MHz. Offset 0..640*

*Selected tuned band determine frequency (kHz or MHz).  
DAB: frequency/BlockNumber = Bitfield:*

*Bit 0 .. 4:*

*L-Band Canada: Numeric value (1 .. 23);*

*L-Band Europe: Numeric value (A=1 .. W=23);*

*Band III: Numeric value (A=1 .. W=23); hex coded*

*Bit 5 .. 8:*

*Band III: Numeric value (not used for L-Band; default value: 0h), hex coded*

*Bit 9:*

*0: Band III, 1: L-Band*

*NOTE: If Station List is 0x5 the parameter Frequency will contain the total number of stations available in a particular PTY.*

*Bit 24-29: HDMulticast*

0x00 – Not Applicable  
0x01 – MC 1

0x02 – MC 2

0x03 – MC 3

0x04 – MC 4

0x05 – MC 5

0x06 – MC 6

0x07 – MC 7

0x08 – PTY List

0x09 – reserved

…

0x3F reserved

*Bit 30: TPStatus:*

0x0: not available

0x1: available

*Bit 31: TMCStatus:*

0x0: not available

0x1: available

*Bit 32-33: reserved*

*Bit 34 - 39: Program Type:*

0x00 – Invalid

0x01 – PTY1  
0x02 – PTY 2  
…

0x1F – PTY31

*Bit 40 up to 312/176 (Coding Table II / Coding Table I): Station Name*

Up to 17 characters 16 letters plus 1 End Of String character

*NOTE: If the Station Name is shorter than 16 characters, it must be terminated with an End Of String character.*

*PSName = 8 Characters Max*

*DAB Service Name = 16 Characters Max*

*DAB Ensemble Name = 16 Characters Max*

*HD Station Name = (SSN)*

*SSN = 4 Characters Max*

*SDARS Channel name = Short name*

*Short Name = 8 Characters Max*

#### TP-LOG-TPL-REQ-023146/A-SID-44-FolderName\_St (TcSE ROIN-146557-2)

Data size: up to 44/24 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x44: NameOfFolder\_St

**Byte 1: Utilization**

0x11: MP\_Media1 – CD

0x12: MP\_Media2 – BT Audio Streaming

0x13: MP\_Media3 – USB

0x15: MP\_Media5 – SD

0x16: MP\_Media6 – DVD

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 43/23 (Coding Table I / Coding Table II): Current folder name**

Max. 20 characters, 19 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023147/A-SID-45-GenreName\_St (TcSE ROIN-146558-2)

Data size: up to 44/24 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x45: NameOfGenre\_St

**Byte 1: Utilization**

0x11: MP\_Media1 – CD

0x12: MP\_Media2 – BT Audio Streaming

0x13: MP\_Media3 – USB

0x14: MP\_Media4 – iPod

0x15: MP\_Media5 – SD

0x16: MP\_Media6 – DVD

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 43/23 (Coding Table I / Coding Table II): Active genre name**

Max. 20 characters, 19 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023148/A-SID-46-TrackName\_St (TcSE ROIN-146559-2)

Data size: up to 44/24 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x46: NameOfTrack\_St

**Byte 1: Utilization**

0x11: MP\_Media1 – CD

0x12: MP\_Media2 – BT Audio Streaming

0x13: MP\_Media3 – USB

0x14: MP\_Media4 – iPod

0x15: MP\_Media5 – SD

0x16: MP\_Media6 – DVD

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 43/23 (Coding Table I / Coding Table II): Current track name**

Max. 20 characters, 19 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023149/A-SID-3B-RadioText\_St (TcSE ROIN-138038-3)

Data size: up to 134/69 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x3B: RadioText\_St

**Byte 1: Utilization**

0x01: Radio\_Service1 – AmFm Radio General

0x03: Radio\_Service3 – DAB

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char) - RDS Latin shall be used.

**Byte 4 up to 133/68: (Coding Table I / Coding Table II): ItemName**

Max. 65 characters, 64 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023150/A-SID-52-GetFolderName\_Rsp (TcSE ROIN-138039-2)

Data size: up to 46/26 (Coding Table I / Coding Table II) bytes.

**Byte 0: Signal identifier**

0x52: GetMPFolderName\_Rsp

**Byte 1: Utilization**

0x11: MP\_Media1 – CD

0x12: MP\_Media2 – BT Audio Streaming

0x13: MP\_Media3 – USB

0x15: MP\_Media5 – SD

0x16: MP\_Media6 – DVD

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4-5: FolderNumber**

0x0000 – Root  
0x0001 – Folder number 1  
0x0002 – Folder number 2  
…  
0xFFFF – Folder number 65535

**Byte 6 up to 45/25 (Coding Table I / Coding Table II): ItemName**

Max. 20 characters, 19 characters plus 1 end of string character.

*NOTE:  
In case of Root, EOS is sent as ItemName, display units to show the HMI defined equivalent of 'Root'.*

#### TP-LOG-TPL-REQ-023151/A-SID-63-GetTagInfo\_Rsp (TcSE ROIN-146549-2)

Data size: up to 1178/668 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x63: GetTagInfo\_Rsp

**Byte 1: Utilization**

0x05: Radio\_Service5 – Radio Tagging

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 1101/556 (Coding Table I / Coding Table II): Tag Info**

*Time Stamp*

*Fix 4 byte*

*Format is binary - ALFN(Time Stamp)*

*Time Lock Status*

*Fix 1 byte*

*0x0 - Time Lock not Set*

*0x1 - Time Lock Set*

*0xFF - (ALFN invalid)*

*Program Number*

*Fix 1 byte*

*0x1 - Multicast 1*

*0x2 - Multicast 2*

*0x3 - Multicast 3*

*0x4 - Multicast 4*

*0x5 - Multicast 5*

*0x6 - Multicast 6*

*0x7 - Multicast 7*

*0xFF - Invalid or Empty*

*Ambiguous Data Flag*

*Fix 1 byte*

*0x00 - Not Ambiguous*

*0x01 - Ambiguous*

*0xFF - Invalid or Empty*

*Button Press Flag*

*Fix 1 byte*

*0x00 - No*

*0x01 - Yes*

*0xFF - Invalid or Empty*

*Note: The following fields shall support Coding Table I & II*

*Title*

*Max. 65 Characters, 64 letters plus 1 EOS*

*If Invalid or Empty set to 0x00*

*Artist*

*Max. 65 Characters, 64 letters plus 1 EOS*

*If Invalid or Empty set to 0x00*

*Album*

*Max. 65 Characters, 64 letters plus 1 EOS*

*If Invalid or Empty set to 0x00*

*UFID Owner Identifier*

*Max. 129 Characters, 128 letters plus 1 EOS*

*If Invalid or Empty set to 0x00*

*UFID Identifier*

*Max. 65 Characters, 64 letters plus 1 EOS*

*If Invalid or Empty set to 0x00*

*Station Call Sign*

*Max. 17 Characters, 16 letters plus 1 EOS*

*If Invalid or Empty set to 0x00*

*Station Frequency*

*Max. 10 Characters, 9 letters plus 1 EOS*

*If Invalid or Empty set to 0x00*

*Genre*

*Max. 129 Characters, 128 letters plus 1 EOS*

*If Invalid or Empty set to 0x00*

#### TP-LOG-TPL-REQ-023152/A-SID-64-GetCDTOCData\_Rsp (TcSE ROIN-146682-2)

Data size: up to 1024 (Coding Table II) bytes

**Byte 0: Signal identifier**

0x64: GetCDTOCData\_Rsp

**Byte 1: Utilization**

0x11: MP\_Media1 – CD

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 1023 (Coding Table II Only): CD TOC Data**

*Array(1..MaxTOC) of record (ItemIndex, TOCItem)*

*Record definition (up to 1020 (Coding Table II) bytes):*

*Byte 0: ItemIndex*

*0x00..0xFF*

*Byte 1 – Byte 3: TOCItem*

*0x00..0xFFFFFF*

#### TP-LOG-TPL-REQ-023153/A-SID-67-DispInfo\_ArtistName\_St (TcSE ROIN-146770-5)

Max Data size: up to 134/69 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x67: DispInfo\_ArtistName\_St

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

0x06: Radio\_Service6 – HD Radio

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**If Utilization = 0x02:**

**Byte 4 up to 93/48 (Coding Table I / Coding Table II):**

AID

Fixed 8 characters

Artist Name

Max. 37 characters, 36 characters plus 1 end of string character.

**If Utilization = 0x06:**

**Byte 4 up to 133/68 (Coding Table I / Coding Table II):**

Artist Name

Max. 65 characters, 64 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023154/A-SID-68-SDARS\_CatName\_St (TcSE ROIN-146772-3)

Data size: up to 56/30 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x68: SDARS\_CatName\_St

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 55/29 (Coding Table I / Coding Table II):**

GCI Category Long Name

Max. 17 characters, 16 characters plus 1 end of string character.

GCI Category Short Name

Max. 9 characters, 8 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023155/A-SID-6C-SDARS\_ChannelName\_St (TcSE ROIN-146811-3)

Data size: up to 64/34 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x6C: SDARS\_ChannelName\_St

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 63/33 (Coding Table I / Coding Table II):**

GCI Channel Long Name

Max. 21 characters, 20 characters plus 1 end of string character.

GCI Channel Short Name

Max. 9 characters, 8 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023156/A-SID-6F-DispInfo\_SongTitle\_St (TcSE ROIN-146818-5)

Max Data size: up to 134/69 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x6F: DispInfo\_SongTitle\_St

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

0x06: Radio\_Service6 – HD Radio

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**If Utilization = 0x02:**

**Byte 4 up to 93/48 (Coding Table I / Coding Table II):**

PID

Fixed 8 characters

Song Title

Max. 37 characters, 36 characters plus 1 end of string character.

**If Utilization = 0x06:**

**Byte 4 up to 133/68 (Coding Table I / Coding Table II):**

Song Title

Max. 65 characters, 64 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023157/A-SID-66-SDARS\_Alert\_St (TcSE ROIN-146819-3)

Data size: up to 95/50 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x66: SDARS\_Alert\_St

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 94/49 (Coding Table I / Coding Table II):**

Alert ID (PID or AID)

Fixed 8 bytes

Channel Number

Fixed 1 Byte

0x00…0xFF

Alert Text (Song Title or Artist Name)

Max. 37 characters, 36 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023158/A-SID-65-SDARS\_SetAlert\_Rq (TcSE ROIN-147030-3)

Data size: up to 1667/927 (Coding Table I/Coding Table II) bytes

**Byte 0: Signal identifier**

0x65: SDARS\_SetAlert\_Rq

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

**Byte 2: Command Execution Status**

0x00: INVALID/INACTIVE

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: OpCode**

*Bit 0-7:OpCode*

0x0: Reserved

0x1: Read

0x2: Add Song

0x3: Add Artist

0x4: Delete

0x5: Delete All

...

0x6..0xFF: Reserved

**Byte 5: NumberOfItems**

0x00: DELETE/DELETE ALL

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

**Byte 6: StartIndex**

0x00: DELETE/DELETE ALL

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

**Byte 7 up to 1666/926 (Coding Table I/Coding Table II): Alert Data**

*Array(1..NumberOfItems) of record (ItemIndex, ID, PDT\_Text)*

*Record definition (up to 1660/920 (Coding Table I/Coding Table II) bytes):*

*Byte 0: ItemIndex*

0x00..0xFF

*Byte 1 to Byte 8: PID/AID*

Fixed 8 Bytes

PID/AID = Max. 8 characters

*Byte 9 up to Byte 82/45: PDT\_Text (Song Title/Song Artist)*

Max. 36 characters plus 1 End Of String

***Note****: Maximum number of alerts that can be added is limited to 20.*

***Notes:***

*IF Opcode = READ, Then IndexNum = 0x00, ID = 0x00, Text = 0x00*

*IF Opcode = ADD SONG, Then IndexNum = StartIndex, ID = PID, Text = PDT Song Title*

*IF Opcode = ADD ARTIST, Then IndexNum = StartIndex, ID = AID, Text = PDT Artist Name*

*IF Opcode = DELETE,*

*Then IndexNum = 0x00, ID = PID/AID, Text = 0x0*

*IF Opcode = DELETE ALL, Then IndexNum = 0x0, ID = 0x0, Text = 0x0*

#### TP-LOG-TPL-REQ-023159/A-SID-6D-SDARS\_CurrentCatList\_Rsp (TcSE ROIN-147071-3)

Data size: up to 3307/1747 (Coding Table I/Coding Table II) bytes

**Byte 0: Signal identifier**

0x6D: SDARS\_CurrentCatList\_Rsp

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: NumberOfItems**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

***Note****: Max number of items returned is limited to 60.*

**Byte 5: StartIndex**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

**Byte 6: ItemsInList**

0x00: Reserved

0x01: Items Available 1

0x02: Items Available 2

....

0xFE: Items Available 254

0xFF: No Entry

**Byte 7 up to 3306/1746 (Coding Table I/Coding Table II): Channel Info**

*Array(1..NumberOfItems) of record (ItemIndex, CategoryNumber, ChannelsInCategory, Short Category Name, Long Category Name)*

*Record definition (up to 55/29 (Coding Table I/Coding Table II) bytes):*

*Byte 0: ItemIndex*

0x00..0xFF

*Byte 1: CategoryNumber:*

0x00: All

0x01: Category 1

…

0xF9: Category 249

0xFA...0xFE: Reserved

0xFF: Invalid

*Byte 2: ChannelsInCategory:*

0x00: Invalid

0x01: Channels Available 1

0x02: Channels Available 2

…

0xFE: Channels Available 254

0xFF: No Entry

***Note:***

*ChannelsInCategory = 0x00:Invalid when there are not channels available in a respective category.*

*ChannelsInCategory = 0xFF:NoEntry for a category number that is out of range.*

*Byte 3 up to Byte 54/28 (Coding Table I/Coding Table II)*

*Short Category Name*

Max. 8 characters plus 1 End Of String

*Long Category Name*

Max. 16 characters plus 1 End Of String

***Note:*** *Both Long and Short Category names will be sent. The HMI will decide which to display.*

#### TP-LOG-TPL-REQ-023160/A-SID-6E-SDARS\_SetAlert\_Rsp (TcSE ROIN-147072-3)

Data size: up to 1668/928 (Coding Table I/Coding Table II) bytes

**Byte 0: Signal identifier**

0x6E: SDARS\_SetAlert\_Rsp

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: RspCode**

0x0: Reserved

0x1: Already Saved

0x2: Memory Full

0x3: List Info

0x4: Added Song

0x5: Added Artist

0x6: Deleted

0x7: All Deleted

...

0x8..0xFF: Reserved

**Byte 5: NumberOfItems**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

***Note****: If RspCode = List Info, then Maximum number of alerts that can be returned is limited to 20.*

**Byte 6: StartIndex**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

**Byte 7: ItemsInList**

0x00: Reserved

0x01: Items Available 1

0x02: Items Available 2

....

0xFE: Items Available 254

0xFF: No Entry

**Byte 8 up to 1667/927 (Coding Table I/Coding Table II): Channel Info**

*Array(1..NumberOfItems) of record (ItemIndex, ID, PDT Text)*

*Record definition (up to 1660/920 (Coding Table I/Coding Table II) bytes):*

*Byte 0: ItemIndex*

0x00..0xFF

*Byte 1 to Byte 8: PID/AID*

Fixed 8 Bytes

Max. 8 characters

*Byte 9 up to Byte 82/45: PDT\_Text (Song Title/Song Artist)*

Max. 36 characters plus 1 End Of String

#### TP-LOG-TPL-REQ-023161/A-SID-62-SSP\_Rsp (TcSE ROIN-147238-2)

Data size: up to 4096 (Coding Table II Only) bytes

**Byte 0: Signal identifier**

0x62: SSP\_Rsp

**Byte 1: Utilization**

0x71: Data\_Service1 – SSP Data Service

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: RspCode**

0x0 - Invalid

0x1 - Data Channel Status

0x2 - decoder\_path\_id

0x3 - SSP Rsp Packet

0x4..0xFF: Reserved

**Byte 5: ChannelStatus**

When RspCode = 0, this parameter = 0x0

When RspCode = 1, this parameter contains the status of the data channel:

0x0 = Initializing channel

0x1 = Channel Closed

0x2 = Channel Open

When RspCode = 2 or 3, this parameter = 0x2.

**Byte 6: decoder\_path\_id**

When RspCode = 0 or 1, this parameter is NULL (0x00)

When RspCode = 0x2 or 0x3, this parameter contains the decoder\_path\_id.

**Byte 7 up to Byte 4095: SSP\_Packet *(Coding Table II Only)***

When RspCode = 0 or 1 or 2, this parameter is NULL (0x00)

When RspCode = 0x3, this parameter contains the SSP response packet.

***Note:*** *SSP packet definitions are defined in the SSP specifications*

#### TP-LOG-TPL-REQ-023162/A-SID-61-SSP\_Rq (TcSE ROIN-147239-3)

Data size: up to 4096 (Coding Table II Only) bytes

**Byte 0: Signal identifier**

0x61: SSP\_Rq

**Byte 1: Utilization**

0x71: Data\_Service1 – SSP Data Service

**Byte 2: Command Execution Status**

0x00: Invalid/Inactive

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: RqCode**

0x0 - Invalid

0x1 - Query Data Channel Status

0x2 - Query decoder\_path\_id

0x3 - SSP Req Packet

0x4..0xFF: Reserved

**Byte 5 up to Byte 4095: SSP\_Packet *(Coding Table II Only)***

When RspCode < 3 this parameter is NULL (0x00)

When RspCode = 0x3, this parameter contains the SSP command packet.

***Note:*** *SSP packet definitions are defined in the SSP specifications*

#### TP-LOG-TPL-REQ-023163/A-SID-80-ChannelInfo\_Rq (TcSE ROIN-167434-2)

Data size: up to 516 (Coding Table III) bytes

**Byte 0: Signal identifier**

0x80: ChannelInfo\_Rq

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

**Byte 2: Command Execution Status**

0x00: INVALID/INACTIVE

**Byte 3: Character Coding**

*Bit 0-5: reserved*

*Bit 6-7: Coding*

0x2: Coding Table III  
0x00-0xFF Hexadecimal Notation

**Byte 4: OpCode**

*Bit 0-7:OpCode*

0x0: Reserved

0x1: Read

0x2: Lock

0x3: Unlock

0x4: Skip

0x5: Clear Skip

0x6: Skip List

0x7: PID Request

...

0x8..0xFF: Reserved

**Byte 5: Category**

0x00: All

0x01: Category1

0x02: Category2

0x03: Category3

…

0xF9: Category249

0xFA: Sirius 1

0xFB: Sirius 2

0xFC: Sirius 3

0xFD: Reserved

0xFE: Reserved

0xFF: Invalid

|  |
| --- |
| ***Note****: If Opcode = Lock/Unlock/Skip/Skip Clear/Skip List, then Category = FF*  *If Opcode = PID Request, then Category = All* |

**Byte 6: StartIndex**

0x00: Invalid

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

|  |
| --- |
| ***Note****: If Opcode = LOCK/UNLOCK/SKIP/SKIP CLEAR/SKIP LIST/PID REQUEST, Then StartIndex = 00* |

**Byte 7: NumberOfItems**

0x00: Invalid

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

|  |
| --- |
| ***Note****: If NumberOfItems exceeds amount of items available, the maximum number of items available will be returned* |

**Byte 8 up to 515 (Coding Table III): Alert Data**

*Array(1..NumberOfItems) of record (Channel Number, Lock Status, Skip Status)*

*Record definition (up to 254 (Coding Table III) bytes):*

*Byte 0: Channel Number*

0x00..0xFF

*Byte 1/Bit 0-3: Lock Status*

*0x00: Invalid*

*0x01: Locked*

*0x02: Unlocked*

Byte 1/Bits 4-7: Skip Status

0x00: Invalid

0x01: Skipped

0x02: Cleared Skip

|  |
| --- |
| Notes:  If Opcode = READ,  Then Channel Number = 0x00, Lock Status = 0x00, Skip Status = 0x00  If Opcode = LOCK,  Then Channel Number = Channel Number, Lock Status = 0x01, Skip Status = 0x00  If Opcode = UNLOCK,  Then Channel Number = Channel Number, Lock Status = 0x02, Skip Status = 0x00  If Opcode = SKIP,  Then Channel Number = Channel Number, Lock Status = 0x00, Skip Status = 0x01  If Opcode = SKIP CLEAR,  Then Channel Number = Channel Number, Lock Status = 0x00, Skip Status = 0x02  If Opcode = SKIP LIST,  Then Channel Number = 0x00, Lock Status = 0x00, Skip Status = 0x00  If Opcode = PID REQUEST,  Then Channel Number = 0x00, Lock Status = 0x00, Skip Status = 0x00 |

#### TP-LOG-TPL-REQ-023164/A-SID-69-SDARS\_ChannelInfo\_Rsp (TcSE ROIN-147031-3)

Data size: up to 3369 (Coding Table II) bytes

**Byte 0: Signal identifier**

0x69: SDARS\_ChannelInfo\_Rsp

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: RspCode**

*Bit 0-7:RspCode*

0x0: Reserved

0x1: List Info

0x2: Locked

0x3: Unlocked

0x4: Skipped

0x5: Skip Cleared

0x6: Skip List

0x7: PID Request

...

0x8..0xFF: Reserved

**Byte 5: Category**

0x00: All

0x01: Category 1

0x02: Category 2

....

0xF9: Category 249

0xFA: Sirius 1

0xFB: Sirius 2

0xFC: Sirius 3

0xFD…0xFF: Reserved

***Note****:*

*If RspCode = Locked/Unlocked/Skipped/Skip Cleared/Skip List, Then Category = FF*

*If RspCode = PID REQUEST, then Category = All*

**Byte 6: NumberOfItems**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

***Note:*** *If RspCode = List Info, Then the max number of items returned is limited to 18.*

***Note:*** *If NumberOfItems exceeds amount of items available, the maximum number of items available will be returned.*

**Byte 7: StartIndex**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

**Byte 8: ItemsInCategory**

0x00: Reserved

0x01: Items Available 1

0x02: Items Available 2

....

0xFE: Items Available 254

0xFF: No Entry

***Note****: If RspCode = Locked/Unlocked/Skipped/Skip Cleared/Skip List, Then ItemsInCategory = FF*

**Byte 9 up to 3368 (Coding Table II): Channel Info**

***Note****: If RspCode = List Info, Then the max number of items returned is limited to 18.*

*Array(1..NumberOfItems) of record (ItemIndex, Channel Number, Lock Status, Skip Status, PID, Short Channel Name, Long Channel Name, Song Artist, Song Title)*

*Record definition (up to 115 (Coding Table II) bytes):*

*Byte 0: ItemIndex*

0x00..0xFF

*Byte 1: Channel Number:*

0x00 ....0xFF

*Byte 2/Bits 0-3: Lock Status:*

0x00: Invalid

0x01: Locked

0x02: Unlocked

*Byte 2/Bits 4-7: Skip Status:*

0x00: Invalid

0x01: Skipped

0x02: Cleared skip

*Byte 3 to Byte 10: PID*

Fixed 8 bytes

8 Characters

*Byte 11 up to 114 (Coding Table II)*

*Short Channel Name*

Max. 8 characters plus 1 End Of String

*Long Channel Name*

Max. 20 characters plus 1 End Of String

***Note:*** *Both Long and Short channel names will be sent. The HMI will decide which to display.*

*Song Artist:*

Max. 36 characters plus 1 End Of String

*Song Title:*

Max. 36 characters plus 1 End Of String

***Notes:***

*If RspCode = List Info, Then*

*Channel Number = Chan. Num.*

*Channel Name = Chan. Name*

*Song Artist = Song Artist*

*Song Title = Song Title*

*Lock Status = Lock Status*

*Skip Status = Skip Status*

*PID = PID*

*If RspCode = Locked, Then*

*Channel Number = Chan. Num.*

*Channel Name = 0x00*

*Song Artist = 0x00*

*Song Title = 0x00*

*Lock Status = 0x01*

*Skip Status = 0x00*

*PID = 0x00*

*If RspCode = Unlocked, Then*

*Channel Number = Chan. Num.*

*Channel Name = 0x00*

*Song Artist = 0x00*

*Song Title = 0x00*

*Lock Status = 0x02*

*Skip Status = 0x00*

*PID = 0x00*

*If RspCode = Skipped, Then*

*Channel Number = Chan. Num.*

*Channel Name = 0x00*

*Song Artist = 0x00*

*Song Title = 0x00*

*Lock Status = 0x00*

*Skip Status = 0x01*

*PID = 0x00*

*If Opcode = Skip Cleared, Then*

*Channel Number = Chan. Num.*

*Channel Name = 0x00*

*Song Artist = 0x00*

*Song Title = 0x00*

*Lock Status = 0x00*

*Skip Status = 0x02*

*PID = 0x00*

*If Opcode = Skip List, Then*

*Channel Number = Chan. Num.*

*Channel Name = 0x00*

*Song Artist = 0x00*

*Song Title = 0x00*

*Lock Status = 0x00*

*Skip Status = Skip Status*

*PID = 0x00*

*If Opcode = PID REQUEST, Then*

*Channel Number = Chan. Num.*

*Channel Name = 0x00*

*Song Artist = 0x00*

*Song Title = 0x00*

*Lock Status = Lock Status*

*Skip Status = Skip Status*

*PID = PID*

#### TP-LOG-TPL-REQ-023165/A-SID-70-AHU\_Bezel\_Diag\_Data (TcSE ROIN-147284-2)

Data size: up to 73 (Coding Table II Only) bytes

**Byte 0: Signal identifier**

0x70: AHU\_Bezel\_Diag\_Data

**Byte 1: Utilization**

0x72: Data\_Service2 – Component Diagnostic Data

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: Bezel\_Diag\_Operation**

0x0 Inactive

0x1 Get All Background Request

0x2 Software Part Number

0x3 Hardware Part Number

0x4 Calibration Part Number

0x5 Speaker Walk-Around

0x6 SDARS ESN Number

0x7 Signal Strength

0x8..0xFF: Reserved

**Byte 5 up to Byte 72: Bezel Diagnostic Data *(Coding Table II Only)***

Max 16 characters + 1 EOS for any Bezel Diagnostic Operation

***Note:*** *When Bezel\_Diag\_Operation = 0x2 then the data will be for the Software Part Number*

*When Bezel\_Diag\_Operation = 0x3 then the data will be for the Hardware Part Number*

*When Bezel\_Diag\_Operation = 0x4 then the data will be for the Calibration Part Number*

*When Bezel\_Diag\_Operation = 0x5 then the data will be for the Speaker Walk-Around test*

*When Bezel\_Diag\_Operation = 0x6 then the data will be for the SDARS ESN Number*

*When Bezel\_Diag\_Operation = 0x7 then the data will be for the radio signal strength test*

**Note:** If Bezel\_Diag\_Operation = 0x1 Get All Background Request then the following diagnostic operation data will be sent in this order:

*Software Part Number*

Max 16 characters + 1 EOS

*Hardware Part Number*

Max 16 characters + 1 EOS

*Calibration Part Number*

Max 16 characters + 1 EOS

*SDARS ESN Number*

Max 16 characters + 1 EOS

#### TP-LOG-TPL-REQ-023166/A-SID-71-EFP\_Bezel\_Diag\_Data (TcSE ROIN-147292-2)

Data size: up to 56 (Coding Table II Only) bytes

**Byte 0: Signal identifier**

0x71: EFP\_Bezel\_Diag\_Data

**Byte 1: Utilization**

0x72: Data\_Service2 – Component Diagnostic Data

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: Bezel\_Diag\_Operation**

0x0 Inactive

0x1 Get All Background Request

0x2 Software Part Number

0x3 Hardware Part Number

0x4 Calibration Part Number

0x5..0xFF: Reserved

**Byte 5 up to Byte 55: Bezel Diagnostic Data *(Coding Table II Only)***

Max 16 characters + 1 EOS for any Bezel Diagnostic Operation

***Note:*** *When Bezel\_Diag\_Operation = 0x2 then the data will be for the Software Part Number*

*When Bezel\_Diag\_Operation = 0x3 then the data will be for the Hardware Part Number*

*When Bezel\_Diag\_Operation = 0x4 then the data will be for the Calibration Part Number*

**Note:** If Bezel\_Diag\_Operation = 0x1 Get All Background Request then the following diagnostic operation data will be sent in this order:

*Software Part Number*

Max 16 characters + 1 EOS

*Hardware Part Number*

Max 16 characters + 1 EOS

*Calibration Part Number*

Max 16 characters + 1 EOS

#### TP-LOG-TPL-REQ-015147/A-SID-72-DSP\_Bezel\_Diag\_Data (TcSE ROIN-147293-2)

Data size: up to 56 (Coding Table II Only) bytes

**Byte 0: Signal identifier**

0x72: DSPAMP\_Bezel\_Diag\_Data

**Byte 1: Utilization**

0x72: Data\_Service2 – Component Diagnostic Data

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: Bezel\_Diag\_Operation**

0x0 Inactive

0x1 Get All Background Request

0x2 Software Part Number

0x3 Hardware Part Number

0x4 Calibration Part Number

0x5 Speaker Walk-Around

0x6..0xFF: Reserved

**Byte 5 up to Byte 55: Bezel Diagnostic Data *(Coding Table II Only)***

Max 16 characters + 1 EOS for any Bezel Diagnostic Operation

***Note:*** *When Bezel\_Diag\_Operation = 0x2 then the data will be for the Software Part Number*

*When Bezel\_Diag\_Operation = 0x3 then the data will be for the Hardware Part Number*

*When Bezel\_Diag\_Operation = 0x4 then the data will be for the Calibration Part Number*

*When Bezel\_Diag\_Operation = 0x5 then the data will be for the Speaker Walk-Around test*

**Note:** If Bezel\_Diag\_Operation = 0x1 Get All Background Request then the following diagnostic operation data will be sent in this order:

*Software Part Number*

Max 16 characters + 1 EOS

*Hardware Part Number*

Max 16 characters + 1 EOS

*Calibration Part Number*

Max 16 characters + 1 EOS

#### TP-LOG-TPL-REQ-023167/A-SID-73-SDARS\_ESN\_St (TcSE ROIN-159079-2)

Data size: up to 28/16 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x73: SDARS\_ESN\_St

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 27/15 (Coding Table I / Coding Table II):**

ESN

Fixed 12 bytes

#### TP-LOG-TPL-REQ-023168/A-SID-74-TMCData\_St (TcSE ROIN-159081-4)

Data size: up to 26 byte

**Byte 0: Signal identifier**

0x74: TMCData\_St

**Byte 1: Utilization**

0x73: Data\_Service3 – TMC Data

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

Bit 0-5: Reserved

Bit 6-7: Coding

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: NbrOfGroups**

Bit 0-4: Reserved

Bit 5-7: NbrOfGroups

0x0: not used

0x1 – 0x5: NbrOfGroups

0x6 – 0x7: Reserved

**Byte 5: Dynamization Information**

Bit 0-2: Duration Persistence

0x0 – 0x7: Numeric Duration Code

Bit 3: Diversion Advice

0x0: no diversion recommended

0x1: diversion recommended

Bit 4: Direction

0x0: positive

0x1: negative

Bit 5-7: Extent

0x0 – 0x7: Numeric Extent Code

**Byte 6-7: Event**

Bit 0-4: Reserved

Bit 5-15: Event Code

0x000 – 0x7FF: Numeric Event Code

**Byte 8-9: Location**

0x0000 – 0xFFFF: Numeric Location Code

**Byte 10 up to 25: OptMsgContent**

Array (2.. NbrOfGroups) of Record (Y-FreeFormat, Z-FreeFormat)

Record definition (4 byte)

Byte 0-1: Y-FreeFormat:

Bit 0-3: Reserved

Bit 4-15: Y11 up to Y0 Free Format

Byte 2-3: Z-FreeFormat:

Bit 0-15: Z15 up to Z0 Free Format

Free Format:

The Free Format used within the OptMsgContent Array must be filled with data as described in the ISO-14819-1 TMC specification.

#### TP-LOG-TPL-REQ-023169/B-SID-76-LBP1\_ItemInfo\_Rsp (TcSE ROIN-159709-6)

Data size: up to Variable (Coding Table I/Coding Table II) bytes

**Byte 0: Signal identifier**

0x76: LBP1\_ItemInfo\_Rsp

**Byte 1: Utilization**

0x01 Radio\_Service1 – AmFm Radio General (AM, FM, AST)

0x02 Radio\_Service2 – SDARS

0x03 Radio\_Service3 – DAB

0x11 MP\_Media1 – CD

0x12 MP\_Media2 – BT Audio Streaming

0x13 MP\_Media3 – USB

0x14 MP\_Media4 – iPod

0x17 MP\_Media7 – Generic Metadata

0x22 Nav\_Service2 – Navigation

0x31 MobileCom\_Service1 – Mobile Phone

0x74: DataService4 – List Browser Data

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: OpCodeRsp:**

*Bit 0-5: reserved*

*Bit 6 - 7: OpCodeRsp*

0x0: Inactive

0x1: GetItemInfoRsp

0x2: SetItemInfoRsp

0x3: Reserved

**Byte 5: *RspListServ :***

0x00: Inactive

0x01: ServerID\_1

...

0xFF: Reserved

**Byte 6-7: ActiveListID**

0x0000: Root

0x0001: ListID\_1

0x0002: ListID\_2

....

0xFFFE:

0xFFFF: Reserved

**Byte 8-9: ParentListID**

0x0000: Root

0x0001: ListID\_1

0x0002: ListID\_2

....

0xFFFE:

0xFFFF: Reserved

**Byte 10: NbrOfItemsRtn**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: Reserved

**Byte 11-12: NbrOfItemsInSelection**

0x0000: Reserved

0x0001:

0x0002:

....

0xFFFE:

0xFFFF:

**Byte 13 up to Variable (Coding Table I/Coding Table II): Channel Info**

*Array(1..NumberOfItemsRtn) of record (ItemIndex, DataType, ObjectType, ObjectState, ActivationEvent, ItemDescriptor)*

*Record definition (up to Variable (Coding Table I/Coding Table II) bytes):*

*Byte 0-1: ItemIndex*

0x0000: List Label

0x0001

..

0xFFFF

*Byte 2: DataType*

0x00

0x01

..

0xFF

*Byte 3:*

Bit 0 - 1: Reserved

Bit 2 - 3: *ObjectType*

0x0: List Label

0x1: Entry

0x2: List

Bit 4 - 5: *ObjectState*

0x0: Inactive

0x1: Active

Bit 6 - 7: *ActivationEvent*

0x0: Not Supported

0x1: Supported

*Byte 4 up to Byte Variable: ItemDescriptor*

{Descriptor Tag} – Refer to descriptor table and DataType.

#### TP-LOG-TPL-REQ-023170/A-SID-20-StreetName\_St (TcSE ROIN-138045-3)

Data size: up to 45/24 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x20: StreetName\_St

**Byte 1: Attribute**

Bit 0-5: reserved

*Bit 6 - 7: Text alignment*

0x0 – centered

0x1 – left aligned

0x2 – right aligned

*NOTE:*

*The text alignment bit can only be used for Gen2 systems*

**Byte 2: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 3 up to 44/23 (Coding Table I / Coding Table II): StreetName**

Max. 21 characters, 20 characters plus 1 end of string character

#### TP-LOG-TPL-REQ-023171/A-SID-0D-Initiate\_BTCall\_Rq (TcSE ROIN-138053-3)

Data size: up to 27 byte.

**Byte 0: Signal identifier**

0x0D: InitiateBTCall\_Rq

**Byte 1: Call Info**

*Bit 0-4: Reserved*

*Bit 5 - 7: TypeOfCall*

0x1 – Telephony Call

0x2 – Last Incoming Call

0x3 – Last Outgoing Call

0x4 – Last Missed Call

0x5 – Redial**Byte 2 up to 26: TelephoneNumber Coding Table II fixed**

Only sent if TypeOfCall = Telephony Call

Max. 25 characters, 24 characters plus 1 end of string.

#### TP-LOG-TPL-REQ-023172/A-SID-78-CurrentStreetName\_St (TcSE ROIN-160690-3)

Data size: up to 46/26 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x78: CurrentStreetName\_St

**Byte 1: Utilization**

0x22: Nav\_Service2 – Navigation

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 45/25 (Coding Table I / Coding Table II):**

*Byte 1:*

*Bits 0-3: Reserved*

*Bits 4 - 7: DataUpdate*

0x0 Inactive

0x1 Set Operation

0x2 Data refresh

*Byte 2: SpeedLimit*

0x00 Invalid

0x01 1

...

0xFF 255

Byte 3: *CurentStreetName*

19 characters max plus 1 end of string character

#### TP-LOG-TPL-REQ-023173/B-SID-77-Destination\_Info\_St (TcSE ROIN-160691-3)

Data size: up to 49/29 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x77: Destination\_Info\_St

**Byte 1: Utilization**

0x22: Nav\_Service2 – Navigation

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 48/28 (Coding Table I / Coding Table II):**

*Byte 1:*

*Bits 0-3:* Reserved

*Bits 4-7:*  *DistUnits*

0x0 Miles

0x1 Kilometres

*Bytes 2-3: TotalDistTraveled*

0x0

...

0xFFFF

*Bytes 4-5: TotalTime* : units=minutes

0x0 0 min

...

0xFFFF 65535 min

*Byte 6: Destination*

19 Characters Max plus 1 end of string character.

#### TP-LOG-TPL-REQ-023174/A-SID-79-MediaInformation\_St (TcSE ROIN-160692-2)

Data size: up to 127/67 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x79: MediaInformation\_St

**Byte 1: Utilization**

0x17: MP\_Media7 – Generic Metadata

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 126/66 (Coding Table I / Coding Table II):**

*Byte 1:*

*Bits 0-2: Reserved*

*Bits 3 - 5: DataUpdate*

0x0 Inactive

0x1 Set Operation

0x2 Data refresh

*Bits 6 - 7: NonMetadataSrc*

0x0 No

0x1 Yes

*Byte 2: MetadataIcon\_1*

0x00 Invalid

0x01.. 0x18 IconID's

0x19 - 0xFF Reserved

*Byte 3: MetadataIcon\_2*

0x00 Invalid

0x01.. 0x18 IconID's

0x19 - 0xFF Reserved

*Byte 4:*

*Metatdata1*

Metadata1

19 Characters Max plus 1 end of string character

*Metadata2*

Metadata2

19 Characters Max plus 1 end of string character

*SourceInformation*

SourceInformation

19 Characters Max plus 1 end of string character

#### TP-LOG-TPL-REQ-023175/A-SID-50-BTCallerIdentification\_St (TcSE ROIN-160784-3)

Data size: up to 66/48 (Coding Table I / Coding Table II) bytes.

**Byte 0: Signal identifier**

0x50 : BTCallerIdentification

**Byte 1: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 2: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 3: BTDeviceIndex**

*Bit 0-3: Reserved*

*Bit 4 - 7: BTDevice\_Index*

0x0 – Reserved

0x1 – BT device index 1

...

0xF – BT device index 15

**Byte 4: Status**

*Bit 0-1: Reserved*

*Bit 2-4: Phone Type*

0x0 - No category

0x1 - Home

0x2 - Office

0x3 - Mobile

0x4 - Other

0x5 – Unknown

0x6 - Fax

*Bit 5-7: Validity*

0x0 – CLID Incoming call available

0x1 – CLID Second incoming call available

0x2 – CLID Outgoing call

0x3 - CLID Incoming SMS Available

0x4 - CLID Incoming Not available

0x5 - CLID Incoming SMS Not available

**Byte 5 up to 65/47 (Coding Table I / Coding Table II):**

**CallID number Coding Table II fixed**

Max. 25 characters, 24 characters plus 1 end of string character.

**CallID Name**

Max. 18 characters, 17 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023176/A-SID-7A-TMCServiceProvider\_St (TcSE ROIN-178778-3)

Data size: 9 byte

**Byte 0: Signal identifier**

0x7A: TMCServiceProvider\_St

**Byte 1: Utilization**

0x73: Data\_Service3 – TMC Data

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

Bit 0-5: Reserved

Bit 6-7: Coding

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: Country Code**

Bit 0-3: Reserved

Bit 4-7: CC

0x00 – 0x0F: Country Code

**Byte 5: Service Identifier**

Bit 0-1: Reserved

Bit 2-7: SID

0x00 – 0x3F: Service Identifier

**Byte 6: Location Table Number**

Bit 0-1: Reserved

Bit 2-7: LTN

0x00 – 0x3F: Location Table Number

**Byte 7: Encryption Information**

Bit 0: Reserved

Bit 1-2: Test mode

0x0: Location code not encrypted

0x1: Location code encrypted

0x2: Reserved

0x3: Full encryption

Bit 3-7: ENCID

0x00 – 0x1F: Encryption Identifier

**Byte 8: Location Table Number (before encryption)**

Bit 0-1: Reserved

Bit 2-7: LTNBE

0x00 – 0x3F: Location Table Number before encryption

Test mode:  
If Test mode is set to "0x00: Location code not encrypted" the terminal shall ignore the ENCID and instead use encryption parameters with values 0,0,0.

If Test mode is set to "0x01: Location code encrypted" the terminal shall ignore ENCID and instead use encryption parameters pre-advised by the service provider (Which of course must be 'pre-stored' within the terminal).

#### TP-LOG-TPL-REQ-023177/A-SID-7B-TMCGetServiceProvider\_Rq (TcSE ROIN-180163-4)

Data size: 20 byte

**Byte 0: Signal identifier**

0x7B: TMCGetServiceProvider\_Rq

**Byte 1: Utilization**

0x73: Data\_Service3 – TMC Data

**Byte 2: Character Coding**

Bit 0-5: Reserved

Bit 6-7: Coding

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 3: CC**

Bit 0-3: Reserved

Bit 4-7: CC

0x00 – 0x0F: Country Code

**Byte 4-11: Preferred Service Provider**

Byte 4:

Bit 7:

0x0 = SID 0x00 is not preferred

0x1 = SID 0x00 is preferred

Byte 4:

Bit 6:

0x0 = SID 0x01 is not preferred

0x1 = SID 0x01 is preferred

…

Byte 11:

Bit 0:

0x0 = SID 0x3F is not preferred

0x1 = SID 0x3F is preferred

**Byte 12-19: Supported Location Table Number**

Byte 12:

Bit 7:

0x0 = LTN 0x00 is not supported

0x1 = LTN 0x00 is supported

Byte 12:

Bit 6:

0x0 = LTN 0x01 is not supported

0x1 = LTN 0x01 is supported

…

Byte 19:

Bit 0:

0x0 = LTN 0x3F is not supported

0x1 = LTN 0x3F is supported

#### TP-LOG-TPL-REQ-023178/A-SID-4F-Initiate\_BTCall\_Rsp (TcSE ROIN-162221-2)

Data size: 2 byte

**Byte 0: Signal identifier**

0x4F : InitiateBTCall\_Rsp

**Byte 1: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

Special response codes:

No Service -> CES 0x24 Final Result – Requested command not supported

Network Error -> CES 0x26 Final Result – Connected Device not present

Number invalid -> CES 0x27 Final Result – Feature not supported

Number busy -> CES 0x28 Final Result – List full

#### TP-LOG-TPL-REQ-023179/A-SID-7C-MyKeyReportCardOutput\_Rsp (TcSE ROIN-201379-1)

Data size: up to 43 bytes

**Byte 0: Signal identifier**

0x7C: MyKeyReportCardOutput\_Rsp

**Byte 1: Utilization**

0x75: Data\_Service5 - DataReport

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

Bit 0-5: Reserved

Bit 6-7: Coding

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char

**Byte 4: ReportInfo**

*Bit 0-3: reserved*

*Bit 4-7: ReportRequested*

0x0 – Inactive  
0x1 – January

0x2 – February

0x3 – March

0x4 – April

0x5 – May

0x6 – June

0x7 – July

0x8 – August

0x9 – September

0xA – October

0xB – November

0xC – December

0xD – Day

0xE – NotUsed

0xF – NotUsed

**Byte 5: MaximumSpeed**

0x00 – 0xFF (0- 255)

**Bytes 6-7: FuelEconomy**

0x0000 – 0x03E7 (0-999)

Note: Resolution of this signal is 0.1.

Examples : 0x000E = 1.4, 0x00FF = 25.5

**Byte 8-9: DriveTime**

0x0000 – 0xAE60 (0-44640 minutes)

**Byte 10 up to 23: BuckledPercentage**

Array(1 - 7) of record (MonitoredSeat, BuckledPercentage)

Record definition (2 bytes):

*Byte 0 : MonitoredSeat*

0x00 – DriverSeat

0x01 – PassengerSeat

0x02 – 0x06 reserved

*Byte 1 : Buckledpercentage*

0x00 – 0x64: Percentage

**Byte 24 up to 27: SyncUsage**

Array(1 - 2) of record (SyncFeature, UsagePercentage)

Record definition (2 bytes):

*Byte 0 : SyncFeature*

0x00 – PhoneConnectedNotActive

0x01 – PhoneConnectedActive

*Byte 1 : UsagePercentage*

0x00 – 0x64: Percentage

**Byte 28: NbrOfSpeedIntervals**

Value: 1 up to 7

**Byte 29 up to 42: SpeedInterval**

Array(1 – NbrOfSpeedIntervals) of record (SpeedInterval, SpeedIntervalPercentage)

Record definition (2 bytes):

*Byte 0 : SpeedInterval*

Value: 0x00 – 0xFF

*Byte 1 : SpeedIntervalPercentage*

0x00 – 0x64: Percentage

#### TP-LOG-TPL-REQ-023180/A-SID-81-CabinComfortPreferenceList\_Rsp (TcSE ROIN-223467-1)

Data size: up to 447/237 (Coding Table I/Coding Table II) bytes

**Byte 0: Signal identifier**

0x81: CabinComfortPreferenceList\_Rsp

**Byte 1: Utilization**

0x81: Charge\_Programming\_Sevice1 – Charge Programming

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: NumberOfItems**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

***Note****: The maximum number of cabin comfort preferences that can be returned is limited to 10.*

**Byte 5: StartIndex**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

**Byte 6: ItemsInList**

0x00: Reserved

0x01: Items Available 1

0x02: Items Available 2

....

0xFE: Items Available 254

0xFF: No Entry

**Byte 7 up to 446/236 (Coding Table I/Coding Table II): List Info**

*Array(1..NumberOfItems) of record (ItemIndex, CabinComfortPrefIDNumber, CabinComfortPreferenceName)*

*Record definition (up to 440/230 (Coding Table I/Coding Table II) bytes):*

*Byte 0: ItemIndex*

0x00: Reserved

0x01: Index1

...

0xFF: Index255

*Byte 1: CabinComfortPrefIDNumber:*

0x00: Null

0x01: Cabin Comfort ID1

0x02: Cabin Comfort ID2

…

0x0A: Cabin Comfort ID10

0x0B: Reserved

...

0xFF: Reserved

*Byte 2 up to Byte 43/22 (Coding Table I/Coding Table II)*

*CabinComfortPreferenceName*

Max. 20 characters plus 1 End Of String

#### TP-LOG-TPL-REQ-023181/B-SID-82-ChargeProfileList\_Rq (TcSE ROIN-223468-1)

Data size: up to 37 (Coding Table III) bytes

**Byte 0: Signal identifier**

0x82: ChargeProfileList\_Rq

**Byte 1: Utilization**

0x81: Charge\_Programming\_Sevice1 – Charge Programming

**Byte 2: Command Execution Status**

0x00: INVALID/INACTIVE

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x2: Coding Table III

0x00-0xFF Hexadecimal Notation

**Byte 4: OpCode**

0x00: Reserved

0x01: Read

0x02: Modify

0x03: Reserved

…

0xFE: Reserved

0xFF: No Entry

**Byte 5: NumberOfItems**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

***Note****: The Maximum number of charge locations that can be returned is limited to 10.*

*If RspCode = MODIFY, then NumberOfItems = 0x01*

**Byte 6: StartIndex**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

***Note****: If RspCode = MODIFY, then StartIndex = 0x01*

**Byte 7 up to 36 (Coding Table III): List Info**

*Array(1..NumberOfItems) of record (ItemIndex, ChargeLocationIDNumber, ChargeLocationName)*

*Record definition (up to 30 (Coding Table I/Coding Table II) bytes):*

*Byte 0: ItemIndex*

0x00: Reserved

0x01: Index1

...

0xFF: Index255

*Byte 1: ChargeProfileIDNumber:*

0x00: Unknown/Any Location

0x01: Location 1

0x02: Location 2

…

0x09: Location 9

0x0A: Reserved

...

0xFF: Reserved

*Byte 2: ChargeProfileChargePreference*

0x00: Null

0x01: ChargeNow

0x02: ValueCharge

0x03: Reserved

…

0xFF: Reserved

|  |
| --- |
| **Notes**:  If OpCode = READ, Then  Byte 7 = 0x00  If OpCode = MODIFY, Then  ItemIndex = ItemIndex  ChargeProfileIDNumber = ChargeProfileIDNumber  ChargeProfileChargePreference = ChargeProfileChargePreference |

#### TP-LOG-TPL-REQ-023182/A-SID-83-ChargeProfileList\_Rsp (TcSE ROIN-223469-2)

Data size: up to 458/248 (Coding Table I/Coding Table II) bytes

**Byte 0: Signal identifier**

0x83: ChargeProfileList\_Rsp

**Byte 1: Utilization**

0x81: Charge\_Programming\_Sevice1 – Charge Programming

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: RspCode**

0x00: Reserved

0x01: List Info

0x02: Modified

0x03: Reserved

…

0xFE: Reserved

0xFF: No Entry

**Byte 5: NumberOfItems**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

***Note****: The Maximum number of charge locations that can be returned is limited to 10*

*If RspCode = Modified, then NumberOfItems = 0xFF*

**Byte 6: StartIndex**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

***Note****: If RspCode = Modified, then StartIndex = 0xFF*

**Byte 7: ItemsInList**

0x00: Reserved

0x01: Items Available 1

0x02: Items Available 2

....

0xFE: Items Available 254

0xFF: No Entry

***Note****: If RspCode = Modified, then itemsInList = 0xFF*

**Byte 8 up to 457/247 (Coding Table I/Coding Table II): List Info**

*Array(1..NumberOfItems) of record (ItemIndex, ChargeLocationIDNumber, ChargeLocationName)*

*Record definition (up to 450/240 (Coding Table I/Coding Table II) bytes):*

*Byte 0: ItemIndex*

0x00: Reserved

0x01: Index1

...

0xFF: Index255

*Byte 1: ChargeProfileIDNumber:*

0x00: Unknown/Any Location

0x01: Location 1

0x02: Location 2

…

0x09: Location 9

0x0A: Reserved

...

0xFF: Reserved

*Byte 2: ChargeProfileChargePreference*

0x00: Null

0x01: ChargeNow

0x02: ValueCharge

0x03: Reserved

…

0xFF: Reserved

*Byte 3 up to Byte 44/23 (Coding Table I/Coding Table II)*

*ChargeLocationName*

Max. 20 characters plus 1 End Of String

|  |
| --- |
| **Notes**:  If RspCode = List Info, Then  ItemIndex = ItemIndex  ChargeProfileIDNumber = ChargeProfileIDNumber  ChargeProfileChargePreference = ChargeProfileChargePreference  If RspCode = Modified, Then  Byte 8 = 0x00 |

#### TP-LOG-TPL-REQ-023183/A-SID-84-ChargeSchedule\_Rq (TcSE ROIN-223470-2)

Data size: up to 56 (Coding Table III) bytes

**Byte 0: Signal identifier**

0x84: ChargeSchedule\_Rq

**Byte 1: Utilization**

0x81: Charge\_Programming\_Sevice1 – Charge Programming

**Byte 2: Command Execution Status**

0x00: INVALID/INACTIVE

**Byte 3: Character Coding**

*Bit 0-5: reserved*

*Bit 6-7: Coding*

0x2: Coding Table III  
0x00-0xFF Hexadecimal Notation

**Byte 4: OpCode**

0x00: Reserved

0x01: Read

0x02: Modify

0x03: Reserved

…

0xFE: Reserved

0xFF: No Entry

**Byte 5: ScheduleType**

0x00: Weekly

0x01: Daily

0x02: Weekday/Weekend

0x03: Reserved

…

0xFF: Reserved

**Byte 6: NumberOfItems**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

***Note:*** *The number of items requested is defined by the schedule type as follows:*

*ScheduleType = Weekly, NumberOfItems = 7*

*ScheduleType = Dailly, NumberOfItems = 1*

*ScheduleType = Weekday/Weekend, NumberOfItems = 2*

**Byte 7 up to 55 (Coding Table III): Charge Schedule Info**

*Array(1..NumberOfItems) of record (BinNumber, ReadyToGo1\_TimeHr, ReadyToGo1\_TimeMin, ReadyToGo1\_CabinComfPrefID, ReadyToGo2\_TimeHr, ReadyToGo2\_TimeMin, ReadyToGo2\_CabinComfPrefID)*

*Record definition (up to 49 (*Coding Table III*) bytes):*

*Byte 0: BinNumber:*

0x00: Null

0x01: Bin 1

0x02: Bin 2

…

0x0A: Bin 10

0x0B: Reserved

...

0xFF: Reserved

*Byte 1: ReadyToGo1\_TimeHr:*

0x00: 0

0x01: 1

0x02: 2

…

0x17: 23

0x18: Reserved

…

0xFE: Reserved

0xFF: Invalid

***Note:*** *Times are always encoded in 24 hour notation.*

*Byte 2: ReadyToGo1\_TimeMin:*

0x00: 0

0x01: 1

0x02: 2

…

0x3B: 59

0x3C: Reserved

…

0xFE: Reserved

0xFF: Invalid

*Byte 3: ReadyToGo1\_CabinComfPrefID:*

0x00: Reserved

0x01: Cabin Comfort ID1

0x02: Cabin Comfort ID 2

…

0x0A: Cabin Comfort ID10

0x0B: Reserved

…

0xFE: Reserved

0xFF: Invalid

*Byte 4: ReadyToGo2\_TimeHr:*

0x00: 0

0x01: 1

0x02: 2

…

0x17: 23

0x18: Reserved

…

0xFE: Reserved

0xFF: Invalid

***Note:*** *Times are always encoded in 24 hour notation*

*Byte 5: ReadyToGo2\_TimeMin:*

0x00: 0

0x01: 1

0x02: 2

…

0x3B: 59

0x3C: Reserved

…

0xFE: Reserved

0xFF: Invalid

*Byte 6: ReadyToGo2\_CabinComfPrefID:*

0x00: Reserved

0x01: Cabin Comfort ID1

0x02: Cabin Comfort ID 2

…

0x0A: Cabin Comfort ID10

0x0B: Reserved

…

0xFE: Reserved

0xFF: Invalid

|  |
| --- |
| **Notes**:  If OpCode = READ, Then  Byte 7 = 0x00  If OpCode = MODIFY, Then  ItemIndex = ItemIndex  BinNumber = BinNumber  ReadyToGo1\_TimeHr = ReadyToGo1\_TimeHr  ReadyToGo1\_TimeMin = ReadyToGo1\_TimeMin  ReadyToGo1\_CabinComfPrefID = ReadyToGo1\_CabinComfPrefID  ReadyToGo2\_TimeHr = ReadyToGo2\_TimeHr  ReadyToGo2\_TimeMin = ReadyToGo2\_TimeMin  ReadyToGo2\_CabinComfPrefID = ReadyToGo2\_CabinComfPrefID |

#### TP-LOG-TPL-REQ-023184/A-SID-85-ChargeSchedule\_Rsp (TcSE ROIN-223471-2)

Data size: up to 85 (Coding Table III) bytes

**Byte 0: Signal identifier**

0x85: ChargeSchedule\_Rsp

**Byte 1: Utilization**

0x81: Charge\_Programming\_Sevice1 – Charge Programming

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x2: Coding Table III  
0x00-0xFF Hexadecimal Notation

**Byte 4: RspCode**

0x00: Reserved

0x01: List Info

0x02: Modified

0x03: Reserved

…

0xFE: Reserved

0xFF: No Entry

**Byte 5: ScheduleType**

0x00: Weekly

0x01: Daily

0x02: Weekday/Weekend

0x03: Reserved

…

0xFF: Reserved

**Byte 6: NumberOfItems**

0x00: Reserved

0x01: 1

0x02: 2

....

0xFE: 254

0xFF: No Entry

***Note:*** *The number of items returned is defined by the schedule type as follows:*

*ScheduleType = Weekly, NumberOfItems = 7*

*ScheduleType = Dailly, NumberOfItems = 1*

*ScheduleType = Weekday/Weekend, NumberOfItems = 2*

*If RspCode = Modified, then NumberOfItems = 0xFF*

**Byte 7: ActiveBin**

0x00: Null

0x01: Bin1

0x02: Bin2

....

0x0A: Bin10

0x0B: Reserved

…

0xFF: Reserved

***Note:*** *If ScheduleType = Daily or Weekday/Weekend, then ActiveBin = 0x00*

**Byte 8 up to 84 (Coding Table III): Charge Schedule Info**

*Array(1..NumberOfItems) of record (ItemIndex, BinNumber, DateDay, DateMonth, DateYear, DayOfWeek, ReadyToGo1\_TimeHr, ReadyToGo1\_TimeMin, ReadyToGo1\_CabinComfPrefID, ReadyToGo2\_TimeHr, ReadyToGo2\_TimeMin, ReadyToGo2\_CabinComfPrefID)*

*Record definition (up to 77 (*Coding Table III*) bytes):*

*Byte 0: BinNumber:*

0x00: Null

0x01: Bin 1

0x02: Bin 2

…

0x0A: Bin 10

0x0B: Reserved

...

0xFF: Reserved

*Byte 1: DateDay:*

0x00: Reserved

0x01: 1

0x02: 2

…

0x1F: 31

0x20: Reserved

…

0xFE: Reserved

0xFF: Invalid

***Note:*** *DateDay = FF when ScheduleType = Daily or Weekday/Weekend.*

*Byte 2: DateMonth*

0x00: Reserved

0x01: January

0x02: February

0x03: March

0x04: April

0x05: May

0x06: June

0x07: July

0x08: August

0x09: September

0x0A: October

0x0B: November

0x0C: December

0x0D: Reserved

…

0xFE: Reserved

0xFF: Invalid

***Note:*** *DateMonth = FF when ScheduleType = Daily or Weekday/Weekend.*

*Byte 3: DateYear:*

0x00: 2010

0x01: 2011

0x02: 2012

…

0x1E: 2040

0x1F: Invalid

0x20: Reserved

…

0xFF: Reserved

***Note:*** *DateYear = FF when ScheduleType = Daily or Weekday/Weekend.*

*Byte 4: DayOfWeek:*

0x00: Reserved

0x01: Sunday

0x02: Monday

0x03: Tuesday

0x04: Wednesday

0x05: Thursday

0x06: Friday

0x07: Saturday

0x08: Reserved

…

0xFE: Reserved

0xFF: Invalid

***Note:*** *DayOfWeek = FF when ScheduleType = Daily or Weekday/Weekend.*

*Byte 5: ReadyToGo1\_TimeHr:*

0x00: 0

0x01: 1

0x02: 2

…

0x17: 23

0x18: Reserved

…

0xFE: Reserved

0xFF: Invalid

***Note:*** *Times are always encoded in 24 hour notation.*

*Byte 6: ReadyToGo1\_TimeMin:*

0x00: 0

0x01: 1

0x02: 2

…

0x3B: 59

0x3C: Reserved

…

0xFE: Reserved

0xFF: Invalid

*Byte 7: ReadyToGo1\_CabinComfPrefID:*

0x00: Reserved

0x01: Cabin Comfort ID1

0x02: Cabin Comfort ID 2

…

0x0A: Cabin Comfort ID10

0x0B: Reserved

…

0xFE: Reserved

0xFF: Invalid

*Byte 8: ReadyToGo2\_TimeHr:*

0x00: 0

0x01: 1

0x02: 2

…

0x17: 23

0x18: Reserved

…

0xFE: Reserved

0xFF: Invalid

***Note:*** *Times are always encoded in 24 hour notation*

*Byte 9: ReadyToGo2\_TimeMin:*

0x00: 0

0x01: 1

0x02: 2

…

0x3B: 59

0x3C: Reserved

…

0xFE: Reserved

0xFF: Invalid

*Byte A: ReadyToGo2\_CabinComfPrefID: :*

0x00: Reserved

0x01: Cabin Comfort ID1

0x02: Cabin Comfort ID 2

…

0x0A: Cabin Comfort ID10

0x0B: Reserved

…

0xFE: Reserved

0xFF: Invalid

|  |
| --- |
| **Notes**:  If RspCode = List Info, Then  ItemIndex = ItemIndex  BinNumber = BinNumber  DateDay = DateDay  DateMonth = DateMonth  DateYear = DateYear  DayOfWeek = DayOfWeek  ReadyToGo1\_TimeHr = ReadyToGo1\_TimeHr  ReadyToGo1\_TimeMin = ReadyToGo1\_TimeMin  ReadyToGo1\_CabinComfPrefID = ReadyToGo1\_CabinComfPrefID  ReadyToGo2\_TimeHr = ReadyToGo2\_TimeHr  ReadyToGo2\_TimeMin = ReadyToGo2\_TimeMin  ReadyToGo2\_CabinComfPrefID = ReadyToGo2\_CabinComfPrefID  If RspCode = Modified, Then  Byte 8 = 0x00 |

#### TP-LOG-TPL-REQ-023185/A-SID-7D-SDARS\_PID\_St (TcSE ROIN-203200-1)

Data size: up to 14 (Coding Table II) bytes

**Byte 0: Signal identifier**

0x7D: SDARS\_PID\_St

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: Response Code**

0x0: Reserved

0x1: New PID

0x2: Clear PID

0x3 - 0xFF: Reserved

**Byte 5 up to 13 (Coding Table II):**

PID

Fixed 8 characters

Channel Number

Fixed 1 Byte

0x00…0xFF

***Note:***

*If RspCode = Clear PID, Then*

*PID = 0x00*

*Channel Number = Channel Number*

#### TP-LOG-TPL-REQ-023186/A-SID-86-SyncSoftwareVersion\_Rsp (TcSE ROIN-229666-1)

Data size: up to 46/25 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x86: SyncSoftwareVersion\_Rsp

**Byte 1: Utilization**

0x81: Charge\_Programming\_Sevice1 – Charge Programming

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 45/24 (Coding Table I / Coding Table II):**

Software Version

Max. 21 characters, 20 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023187/A-SID-87-TelServESN\_St (TcSE ROIN-229665-1)

Data size: up to 30/17 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x87: TelServESN\_St

**Byte 1: Utilization**

0x81: Charge\_Programming\_Sevice1 – Charge Programming

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 29/16 (Coding Table I / Coding Table II):**

ESN

Max. 13 characters, 12 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023188/A-SID-88-TelServUserID\_St (TcSE ROIN-229667-1)

Data size: up to 130/67 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x88: TelServUserID\_St

**Byte 1: Utilization**

0x81: Charge\_Programming\_Sevice1 – Charge Programming

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 129/66 (Coding Table I / Coding Table II):**

Old User

Max. 21 characters, 20 characters plus 1 end of string character.

New User

Max. 21 characters, 20 characters plus 1 end of string character.

Current User

Max. 21 characters, 20 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-023189/A-SID-89-ConsHistGraph\_St (TcSE ROIN-266595-1)

Data size: up to 35 bytes.

**Byte 0: Signal identifier**

0x89: ConsHistGraph\_St

**Byte 1: Utilization**

0x82: Electrification Information

**Byte 2: Command Execution Status**

0x0y: Final Result - Success

0x1y: Final Result - Fail

0x2y: Final Result - Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x2: Coding Table III

0x00-0xFF: Hexadecimal Notation

**Byte 4: ConsHist1\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 5: ConsHistThr1\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 6: ConsHistColor1\_D\_Dsply**

0x0: Color1

0x1: Color2

0x2: Grayed\_Out

**Byte 7: ConsHist2\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 8: ConsHistThr2\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 9: ConsHistColor2\_D\_Dsply**

0x0: Color1

0x1: Color2

0x2: Grayed\_Out

**Byte 10: ConsHist3\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 11: ConsHistThr3\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 12: ConsHistColor3\_D\_Dsply**

0x0: Color1

0x1: Color2

0x2: Grayed\_Out

**Byte 13: ConsHist4\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 14: ConsHistThr4\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 15: ConsHistColor4\_D\_Dsply**

0x0: Color1

0x1: Color2

0x2: Grayed\_Out

**Byte 16: ConsHist5\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 17: ConsHistThr5\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 18: ConsHistColor5\_D\_Dsply**

0x0: Color1

0x1: Color2

0x2: Grayed\_Out

**Byte 19: ConsHist6\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 20: ConsHistThr6\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 21: ConsHistColor6\_D\_Dsply**

0x0: Color1

0x1: Color2

0x2: Grayed\_Out

**Byte 22: ConsHist7\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 23: ConsHistThr7\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 24: ConsHistColor7\_D\_Dsply**

0x0: Color1

0x1: Color2

0x2: Grayed\_Out

**Byte 25: ConsHist8\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 26: ConsHistThr8\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 27: ConsHistColor8\_D\_Dsply**

0x0: Color1

0x1: Color2

0x2: Grayed\_Out

**Byte 28: ConsHist9\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 29: ConsHistThr9\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 30: ConsHistColor9\_D\_Dsply**

0x0: Color1

0x1: Color2

0x2: Grayed\_Out

**Byte 31: ConsHist10\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 32: ConsHistThr10\_Pc\_Dsply**

0x00: 0

0x01: 1

…

0x7D: 125

0x7E: Data\_Not\_Available

0x7F: Faulty

**Byte 33: ConsHistColor10\_D\_Dsply**

0x0: Color1

0x1: Color2

0x2: Grayed\_Out

**Byte 34: ConsHistTIPC\_D\_Dsply**

0x0: Invalid

0x1: Display\_1\_Minute\_Increments

0x2: Display\_2\_Minute\_Increments

0x3: Display\_6\_Minute\_Increments

**Byte 35: ConsUnitIPC\_D\_Dsply**

0x0: Invalid

0x1: Miles\_Gallon\_US

0x2: Liters\_100\_km

0x3: km\_L

0x4: miles\_gallon\_UK

0x5: Whr\_mi

0x6: Whr\_km

0x7: Miles\_Gallon

0x8: Liters\_100\_km\_equivalent

0x9: km\_L\_equivalent

0xA: miles\_gallon\_UK\_equivalent

0xB: Reserved\_11

0xC: Reserved\_12

0xD: Reserved\_13

0xE: Reserved\_14

0xF: Reserved\_15

#### TP-LOGv2-TPL-REQ-013859/A-SID-22-NavigationSymbolInfo\_St (TcSE ROIN-281274)

Data size up to 46 bytes

**Byte 0: Signal identifier**

0x22: NavigationSymbolInfo\_St

**Byte 1: HeaderInfo**

*Bit 0–4: reserved*

*Bit 5-6: UnitOfLength*

0x0 – kilometres

0x1 – miles

0x2 – metres

0x3 – yards

*Bit 7: PropertyOfDistance*

0x0 – bargraph

0x1 – length

*NOTE:*

*The values 0x2 and 0x3 of the UnitOfLength shall only be used for Gen3 systems*

**Byte 2 -3: DistanceToNextManeuver**

Values: 0x0 up to 0xFFFF

*NOTE:*

*If the "PropertyOfDistance" is set to "length" the "DistanceToNextManeuver" will be in steps 0.1 kilometres / miles.*

*If the "Property of distance" is set to "bargraph" the "DistanceToNextmaneuver" will be in steps 0.01 kilometres / miles.*

*If the "UnitOfLength" is set metres or yards the "DistanceToNextManeuver" will be in steps of 5.*

**Byte 4: BargraphSteps**

Values: 0x00 up to 0xFF

*NOTE:*The relative size of bargraph (0% – 100% = 0x00 – 0xFF). The BargraphSteps value decreases from 0xFF at start down to 0x00 when the decision point is reached.

**Byte 5: NumberOfStreetSegments**

Value: 0x1 up to 0x14

**Byte 6 - 45: StreetSegments**

Array(1..NumberOfStreetSegments) of record (DirectionAndNumber, ManeuverElement)

*Bit 0-7: DirectionAndNumber*

0x00 – North

0x01 – 1

0x02 – 2

0x03 – 3

0x04 – 4

0x05 – 5

0x06 – 6

0x07 – 7

0x08 – 8

0x09 – 9

0x10 – North – North – West

0x20 – North – West

0x30 – West – North – West

0x40 – West

0x50 – West – South – West

0x60 – South – West

0x70 – South – South – West

0x80 – South

0x90 – South – South – East

0xA0 – South – East

0xB0 – East – South – East

0xC0 – East

0xD0 – East – North – East

0xE0 – North – East

0xF0 – North – North – East

0xFF – No direction

*Bit 8-15: ManeuverElement*

0x00 – NoSymbol (NO\_SYMBOL)

0x01 – SideStreet ((SIDESTREET)

0x02 – Silent (SILENT)

0x03 – Turn (TURN)

0x04 – UTurnTrafficRightSide (U\_TURN\_TRS\_RIGHT)

0x05 – UTurnTrafficLeftSide (U\_TURN\_TRS\_LEFT)

0x06 – ChangeLane (FILTER)

0x07 – ServiceRoad (PARALLEL\_CWY)

0x08 – ServiceRoad (SERVICE ROAD)

0x09 – Fork (ORIENTATE)

0x0A – Exit (EXIT)

0x0B – TurnOnMainroad (MAINROAD)

0x0C – RoundaboutTrafficRightSide (ROUNDABOUT\_TRS\_RIGHT)

0x0D – RoundaboutTrafficLeftSide (ROUNDABOUT\_TRS\_LEFT)

0x0E – SquareTrafficRightSide (SQUARE\_TRS\_RIGHT)

0x0F – SquareTrafficLeftSide (SQUARE\_TRS\_LEFT)

0x10 – NoInfo (NO\_INFO)

0x11 – FollowStreet (FOLLOW\_STREET)

0x12 – ChangeLane (PREPARE\_TURN)

0x13 – ArrivedAtDestination (DEST\_REACHED)

0x14 - ArrivedAtWaypoint

0x15 - ApproachingDestination

0x16 - ApproachingWaypoint

0x17 - EnterHighway

0x18 - FerryAhead

0x19 - Merge

0x20 – OffRoad (OFF\_ROAD)

0x21 – OffMap (OFF\_MAP)

0x22 – NoRoute (NO\_ROUTE)

0x23 – CalcRoute (CALC\_ROUTE)

0x24 – ArrivedDestinationOffMap (DEST\_AREA)

0x25 – RecalcRoute (RECALC\_ROUTE)

0x30 – Number (NUMBER)

#### TP-LOG-TPL-REQ-023190/A-SID-22-NavigationSymbolInfo\_St (TcSE ROIN-138046-7)

Data size up to 46 bytes

**Byte 0: Signal identifier**

0x22: NavigationSymbolInfo\_St

**Byte 1: HeaderInfo**

*Bit 0–4: reserved*

*Bit 5-6: UnitOfLength*

0x0 – kilometres

0x1 – miles

0x2 – metres

0x3 – yards

*Bit 7: PropertyOfDistance*

0x0 – bargraph

0x1 – length

*NOTE:*

*The values 0x2 and 0x3 of the UnitOfLength shall only be used for Gen3 systems*

**Byte 2 -3: DistanceToNextManeuver**

Values: 0x0 up to 0xFFFF

*NOTE:*

*If the "PropertyOfDistance" is set to "length" the "DistanceToNextManeuver" will be in steps 0.1 kilometres / miles.*

*If the "Property of distance" is set to "bargraph" the "DistanceToNextmaneuver" will be in steps 0.01 kilometres / miles.*

*If the "UnitOfLength" is set metres or yards the "DistanceToNextManeuver" will be in steps of 5.*

*DistanceToNextManeuver shall be sent in INTEL format. For example, if DistanceToNextManuever is 1.2 Miles and PropertyOfDistance = length, Byte 2 = 0C and byte 3 = 00*

**Byte 4: BargraphSteps**

Values: 0x00 up to 0xFF

*NOTE:*The relative size of bargraph (0% – 100% = 0x00 – 0xFF). The BargraphSteps value decreases from 0xFF at start down to 0x00 when the decision point is reached.

**Byte 5: NumberOfStreetSegments**

Value: 0x1 up to 0x14

**Byte 6 - 45: StreetSegments**

Array(1..NumberOfStreetSegments) of record (DirectionAndNumber, ManeuverElement)

*Bit 0-7: DirectionAndNumber*

0x00 – North

0x01 – 1

0x02 – 2

0x03 – 3

0x04 – 4

0x05 – 5

0x06 – 6

0x07 – 7

0x08 – 8

0x09 – 9

0x10 – North – North – West

0x20 – North – West

0x30 – West – North – West

0x40 – West

0x50 – West – South – West

0x60 – South – West

0x70 – South – South – West

0x80 – South

0x90 – South – South – East

0xA0 – South – East

0xB0 – East – South – East

0xC0 – East

0xD0 – East – North – East

0xE0 – North – East

0xF0 – North – North – East

0xFF – No direction

*Bit 8-15: ManeuverElement*

0x00 – NoSymbol (NO\_SYMBOL)

0x01 – SideStreet ((SIDESTREET)

0x02 – Silent (SILENT)

0x03 – Turn (TURN)

0x04 – UTurnTrafficRightSide (U\_TURN\_TRS\_RIGHT)

0x05 – UTurnTrafficLeftSide (U\_TURN\_TRS\_LEFT)

0x06 – ChangeLane (FILTER)

0x07 – ServiceRoad (PARALLEL\_CWY)

0x08 – ServiceRoad (SERVICE ROAD)

0x09 – Fork (ORIENTATE)

0x0A – Exit (EXIT)

0x0B – TurnOnMainroad (MAINROAD)

0x0C – RoundaboutTrafficRightSide (ROUNDABOUT\_TRS\_RIGHT)

0x0D – RoundaboutTrafficLeftSide (ROUNDABOUT\_TRS\_LEFT)

0x0E – SquareTrafficRightSide (SQUARE\_TRS\_RIGHT)

0x0F – SquareTrafficLeftSide (SQUARE\_TRS\_LEFT)

0x10 – NoInfo (NO\_INFO)

0x11 – FollowStreet (FOLLOW\_STREET)

0x12 – ChangeLane (PREPARE\_TURN)

0x13 – ArrivedAtDestination (DEST\_REACHED)

0x14 - ArrivedAtWaypoint

0x15 - ApproachingDestination

0x16 - ApproachingWaypoint

0x17 - EnterHighway

0x18 - FerryAhead

0x19 - Merge

0x20 – OffRoad (OFF\_ROAD)

0x21 – OffMap (OFF\_MAP)

0x22 – NoRoute (NO\_ROUTE)

0x23 – CalcRoute (CALC\_ROUTE)

0x24 – ArrivedDestinationOffMap (DEST\_AREA)

0x25 – RecalcRoute (RECALC\_ROUTE)

0x30 – Number (NUMBER)

#### TP-LOG-TPL-REQ-023191/A-SID-8D-RadioText2\_St (TcSE ROIN-296321-1)

Data size: up to 262/133 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x8D: RadioText2\_St

**Byte 1: Utilization**

0x03: Radio\_Service3 – DAB

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char) - RDS Latin shall be used.

**Byte 4 up to 261 /132: (Coding Table I / Coding Table II): ItemName**

Max. 129 characters, 128 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-048851/A-SID-90-EmergencyCallText\_St (TcSE ROIN-305875-1)

Data size: up to 406/205 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x90: EmergencyCallText\_St

**Byte 1: Utilization**

0x32: MobileCom\_Service2 – Embedded Modem

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 405/204 (Coding Table I / Coding Table II):**

EraGText

Max. 201 characters, 200 characters plus 1 end of string character.

#### TP-LOG-TPL-REQ-092298/A-SID-91-UpcomingStreetName\_St

Data size: up to 66/37 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal identifier**

0x91: UpcomingStreetName\_St

**Byte 1: Utilization**

0x22: Nav\_Service2 – Navigation

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result– Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: Path Index**

*Bit 0-1: Reserved*

*Bit 2-7: Coding*

*0x0 - 7 = Reserved*

*0x8 - 63 = IndexOfPath*

**Byte 5: Stub Path Index**

*Bit 0-1: Reserved*

*Bit 2-7: Coding*

*0x0 = StubStartsFirstPathInTheHorizon*

*0x1 - 7 = Reserved*

*0x8 - 63 = SubIndexOfPath*

**Byte 6: Road Shield Icon:**

*Bit 0-7: Coding*

**Byte 7 up to 24/15 (Coding Table I / Coding Table II): Road Shield Text:**

Byte 1 up to 18/9 (Coding Table I / Coding Table II): RoadShieldText

9 characters, 8 letters plus 1 end of string character

**Byte 25/16 up to 65/36 (Coding Table I / Coding Table II): Upcoming Street Name:**

Byte 1 up to 40/20 (Coding Table I / Coding Table II): UpcomingStreetName

20 characters, 19 letters plus 1 end of string character

#### TP-LOG-TPL-REQ-023249/B-SID-92-DynamicLabelPlus\_St (TcSE ROIN-286211)

Data size: up to 520/391 (Coding Table I / Coding Table II) byte

**Byte 0: Signal identifier**

0x92: DynamicLabelPlus\_St

**Byte 1: Utilization**

0x03: Radio\_Service3 – DAB

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char) (RDS Latin used instead)

**Byte 4: ItemInfo**

*Bit 0-5: Reserved*

*Bit 6-7: CommandTypeCode*

0x0: New

0x1: Update

0x2: Delete

0x3: Reserved

**Byte 5: NbrOfTypes**

*Bit 0: Reserved*

*Bit 1-7: NbrOfTypes*

*0x00: NoTagsAvailable*

*0x01 up to 0x40*

**Byte 6 up to 263/134 (Coding Table I / Coding Table II): RadioText**

Max. 129 characters, 128 letters plus 1 end of string character

**Byte 264/135 up to 519/390 (Coding Table I / Coding Table II): ItemVector**

Array (1.. NbrOfTypes) of Record (ContentTypeID, ContentType, StartMarker, LengthMarker)

Record definition (4 bytes):

*Byte 1: ContentTypeID*

*0x01 up to 0x40*

*Byte 2: ContentType*

*0x01 up to 0x40*

*Byte 3: StartMarker*

*0x01 up to 0x80*

*Byte 4: LengthMarker*

*0x01 up to 0x80*

#### TP-LOG-TPL-REQ-134551/A-SID-93-JournalineTxtMsg\_St

Data size: up to 4095 (Fixed Coding Table II) byte

**Byte 0: Signal identifier**

0x93: JournalineTxtMsg\_St

**Byte 1: Utilization**

0x03: Radio\_Service3 – DAB

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char) (RDS Latin used instead)

**Byte 4 up to 4094 (Fixed Coding Table II): Text Message:**

Max. 4091 characters, 4090 letters plus 1 end of string character

#### TP-LOG-TPL-REQ-166128/A-SID-94-WifiInfo\_Rq

Data Size: up to 201/103 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal Identifier**

0x94: WifiInfo\_Rq

**Byte 1: Utilization**

0x32: MobileCom\_Service2 - Embedded Modem

**Byte 2: Command Execution Status**

0x00: INVALID/INACTIVE

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: OpCode**

0x00: Reserved

0x01: Read

0x02: WriteSSID

0x03: WritePassword

0x04 - 0xFF: Reserved

**Byte 5 up to 201/103 (Coding Table I / Coding Table II):**

Password

Max. 65 characters, 64 plus 1 end of string

SSID

Max. 33 characters, 32 plus 1 end of string

**Note:**

When OpCode = 0x01 Read, Password and SSID will be end of string (0x00)

When OpCode = 0x02 WriteSSID, Password will be end of string (0x00)

When OpCode = 0x03: WritePassword, SSID will be end of string (0x00)

#### TP-LOG-TPL-REQ-166129/B-SID-95-WifiInfo\_Rsp

Data Size: up to 201/103 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal Identifier**

0x95: WifiInfo\_Rsp

**Byte 1: Utilization**

0x32: MobileCom\_Service2 - Embedded Modem

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: OpCode**

0x00: Reserved

0x01: Data

0x02: SSIDWritten

0x03: PasswordWritten

0x04 - 0xFF: Reserved

**Byte 5 up to 200/102 (Coding Table I / Coding Table II):**

Password

Max. 65 characters, 64 plus 1 end of string

SSID

Max. 33 characters, 32 plus 1 end of string

**Note:**

When OpCode = 0x01 Data, Password and SSID will both be sent

When OpCode = 0x02 SSIDWritten or 0x03 PasswordWritten, Password and SSID will both be end of string (0x00)

#### TP-LOG-TPL-REQ-166130/C-SID-96-CarrierInfo\_Rsp

Data Size: up to 876/440 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal Identifier**

0x96: CarrierInfo\_Rsp

**Byte 1: Utilization**

0x32: MobileCom\_Service2 - Embedded Modem

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 875/439 (Coding Table I / Coding Table II):**

Ford Landing URL

Max. 193 characters, 192 plus 1 end of string

Lincoln Landing URL

Max. 193 characters, 192 plus 1 end of string

Ford Phone Number

Max. 25 characters, 24 plus 1 end of string

Lincoln Phone Number

Max. 25 characters, 24 plus 1 end of string

#### TP-LOG-TPL-REQ-166131/B-SID-97-DataUsage\_Rsp

Data Size: up to 26 (Coding Table III) bytes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Byte 0: Signal Identifier** | | | | | |
|  | | | 0x97: DataUsage\_Rsp | | |
|  | | |  | | |
| **Byte 1: Utilization** | | | | | |
|  | | | 0x32: MobileCom\_Service2 - Embedded Modem | | |
|  | |  | | | |
| **Byte 2: Command Execution Status** | | | | | |
|  | | | 0x0y: Final Result – Success | | |
|  | | | 0x1y: Final Result – Fail | | |
|  | | | 0x2y: Final Result – Information | | |
|  | | | 0x3y: Intermediate Result– Wait | | |
|  | |  | | | |
| **Byte 3: Character Coding** | | | | | |
|  | | | *Bit 0-5: Reserved* | | |
|  | | | *Bit 6-7: Coding* | | |
|  | | | | 0x2: Coding Table III | |
|  | | | | 0x00-0xFF Hexadecimal Notation | |
|  | |  | | | |
| **Byte 4: Counter Hour** | | | | | |
|  | | | 0x00: Hour 0 | | |
|  | | | … | | |
|  | | | 0x17: Hour 24 | | |
|  | | | 0x18 - 0xFE: Reserved | | |
|  | | | 0xFF: Invalid | | |
|  | |  | | | |
| **Byte 5: Counter Minute** | | | | | |
|  | | | 0x00: Minute 0 | | |
|  | | | … | | |
|  | | | 0x3B: Minute 59 | | |
|  | | | 0x3C - 0xFE: Reserved | | |
|  | | | 0xFF: Invalid | | |
|  | |  | | | |
| **Byte 6: Counter Second** | | | | | |
|  | | | 0x00: Second 0 | | |
|  | | | … | | |
|  | | | 0x3B: Second 59 | | |
|  | | | 0x3C - 0xFE: Reserved | | |
|  | | | 0xFF: Invalid | | |
|  | |  | | | |
| **Byte 7: Plan Type** | | | | | |
|  | | | 0x00: Invalid | | |
|  | | | 0x01: Shared | | |
|  | | | 0x02: Session | | |
|  | |  | | | |
| **Byte 8: Expiry/Renewal Date** | | | | | |
|  | | | 0x00: Invalid | | |
|  | | | 0x01: Expiry Date | | |
|  | | | 0x02: Renewal Date | | |
|  | |  | | | |
| **Byte 9: Expiry/Renewal Month** | | | | | |
|  | | | 0x00: Invalid | | |
|  | | | 0x01: January | | |
|  | | | 0x02: February | | |
|  | | | 0x03: March | | |
|  | | | 0x04: April | | |
|  | | | 0x05: May | | |
|  | | | 0x06: June | | |
|  | | | 0x07: July | | |
|  | | | 0x08: August | | |
|  | | | 0x09: September | | |
|  | | | 0x0A: October | | |
|  | | | 0x0B: November | | |
|  | | | 0x0C: December | | |
|  | | | 0x0D - 0xFF: Reserved | | |
|  | |  | | | |
| **Byte 10: Expiry/Renewal Day** | | | | | |
|  | | | 0x00: Invalid | | |
|  | | | 0x01: Day 1 | | |
|  | | | … | | |
|  | | | 0x1F: Day 31 | | |
|  | | | 0x0D - 0xFF: Reserved | | |
|  | |  | | | |
| **Byte 11: Expiry/Renewal Year** | | | | | |
|  | | | 0x00: Year 2000 | | |
|  | | | … | | |
|  | | | 0xFE: Year 2254 | | |
|  | | | 0xFF: Invalid | | |
|  | |  | | | |
| **Byte 12: Expiry/Renewal Hour** | | | | | |
|  | | | 0x00: Hour 0 | | |
|  | | | … | | |
|  | | | 0x17: Hour 24 | | |
|  | | | 0x18 - 0xFE: Reserved | | |
|  | | | 0xFF: Invalid | | |
|  | |  | | | |
| **Byte 13: Expiry/Renewal Minute** | | | | | |
|  | | | 0x00: Minute 0 | | |
|  | | | … | | |
|  | | | 0x3B: Minute 59 | | |
|  | | | 0x3C - 0xFE: Reserved | | |
|  | | | 0xFF: Invalid | | |
|  | |  | | | |
| **Byte 14: Expiry/Renewal Second** | | | | | |
|  | | | 0x00: Second 0 | | |
|  | | | … | | |
|  | | | 0x3B: Second 59 | | |
|  | | | 0x3C - 0xFE: Reserved | | |
|  | | | 0xFF: Invalid | | |
|  | |  | | | |
| **Bytes 15-17: Data Used** | | | | | |
|  | | | 0x000000: Data 0.00 | | |
|  | | | … | | |
|  | | | 0x01869F Data 999.99 | | |
|  | | | 0x0186A0 - 0xFFFFFE: Reserved | | |
|  | | | 0xFFFFFF: Invalid | | |
|  | |  | | | |
| **Note:** | | |  | | |
| Data values are in steps of 0.01 decimal units (where the units used are outlined in Data Used Units) Ex. 0x008707 = 34567 = 345.67 Mb (or Kb or Gb) | | | | | |
|  | |  | | | |
| **Byte 18: Data Used Units** | | | | | |
|  | | | 0x0: Invalid | | |
|  | | | 0x1: Mb | | |
|  | | | 0x2: Kb | | |
|  | | | 0x3: Gb | | |
|  | |  | | | |
| **Bytes 19-21: Total Data** | | | | | |
|  | | | 0x00000: Data 0.00 | | |
|  | | | … | | |
|  | | | 0x01869F Data 999.99 | | |
|  | | | 0x0186A0: Unlimited | | |
|  | | | 0x0186A1 - 0xFFFFFE: Reserved | | |
|  | | | 0xFFFFFF: Invalid | | |
|  | |  | | | |
| **Note:** | | |  | | |
| Data values are in steps of 0.01 decimal units (where the units used are outlined in Total Data Units) Ex. 0x008707 = 34567 = 345.67 Mb (or Kb or Gb) | | | | | |
|  | |  | | | |
| **Byte 22: Total Data Units** | | | | | |
|  | | | 0x0: Invalid | | |
|  | | | 0x1: Mb | | |
|  | | | 0x2: Kb | | |
|  | | | 0x3: Gb | | |
|  | |  | | | |
| **Note:** | | |  | | |
| When Total Data Units = 0x0, no units shall be shown, no characters shall be displayed. | | | | | |
|  | |  | | | |
| **Byte 23: Data Used Percent** | | | | | |
|  | | | 0x00: Data 0% | | |
|  | | | … | | |
|  | | | 0x64: Data 100% | | |
|  | | | 0x65 - 0xFE: Reserved | | |
|  | | | 0xFF: Invalid | | |
|  | |  | | | |
| **Byte 24: Overage Flag** | | | | | |
|  | | | 0x0: Invalid | | |
|  | | | 0x1: No | | |
|  | | | 0x2: Yes | | |
|  | |  | | | |
| **Byte 25: Data Plan Status** | | | | | |
|  | | | 0x00: Invalid | | |
|  | | | 0x01: Free Trial Period Waiting | | |
|  | | | 0x02: Free Trial Period Active | | |
|  | | | 0x03: No Active Subscription | | |
|  | | | 0x04: Active Subscription | | |

#### TP-LOG-TPL-REQ-166132/B-SID-98-DeviceList\_Rsp

Data Size: up to 1427/727 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal Identifier**

0x98: DeviceList\_Rsp

**Byte 1: Utilization**

0x32: MobileCom\_Service2 - Embedded Modem

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: List Type**

0x00: Reserved

0x01: Connected List

0x02: BlockedList

0x03: Reserved

…

0x06: Reserved

0x07: No Entry

**Byte 5: List Size**

0x00: Inactive

0x01: List Size 1

…

0x14: List Size 20

0xFF: No Entry

**Note:**

List Size maximum is 20

**Byte 6: Total Number Of Devices Available**

0x00: Inactive

0x01: 1 Device Available

…

0xFE: 254 Devices Available

0xFF: No Entry

**Byte 7 up to 1426/726 (Coding Table I/ Coding Table II): Vector**

Array (1…N) of record (IndexNumber, DeviceName, MAC) with

TotalNumberOfDevices defined in ListSize

Record definition (up to 1420/720 (Coding Table I/Coding Table II) bytes):

Byte 0: ItemIndex

0x00 Inactive

0x01 Index 1

…

0xFF Index 255

Byte 1 up to 70/35 (Coding Table I/Coding Table II):

MAC

Fixed 17 characters

Device Name

Max. 18 characters, 17 plus 1 end of string

**Note:**

If there are no devices in the list, List Size and Total Number Of Devices Available = 0xFF: No Entry

The Vector Array shall not be transmitted

#### TP-LOG-TPL-REQ-194071/A-SID-99-TrafficServiceProvider\_St

Data size: 8 byte

**Byte 0: Signal identifier**

0x99: TrafficServiceProvider\_St

**Byte 1: Utilization**

0x73: Data\_Service3 – TPEG Data

**Byte 2: Character Coding**

Bit 0-5: Reserved

Bit 6-7: Coding

0x2: Coding Table III

0x00-0xFF RawData (Hexadecimal Notation)

**Byte 3: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 4-6: tuned SID**

0x000000: No SID tuned

0x000001 – 0xFFFFFF: SID

**Byte 7: Service Status**

Bit 0-4: Reserved

Bit 5 PreferredSIDStatus:

0x0: Preferred SID not available

0x1: Preferred SID available

Bit 6 SupportedSIDStatus:

0x0: Supported SID not available

0x1: Supported SID available

Bit 7 SIDTableEmpty

0x0 inactive

0x1 active

#### TP-LOG-TPL-REQ-194072/A-SID-9A-TrafficGetServiceProvider\_Rq

Data size: up to 197 byte

**Byte 0: Signal identifier**

0x9A: TrafficGetServiceProvider\_Rq

**Byte 1: Utilization**

0x73: Data\_Service3 – TPEG Data

**Byte 3: Character Coding**

Bit 0-5: Reserved

Bit 6-7: Coding

0x2: Coding Table III

0x00-0xFF RawData (Hexadecimal Notation)

**Byte 3: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 4-6: preferred SIDs**

preferredSID:

0x000000 – 0xFFFFFF

**Byte 7: NbrOfSupportedSIDs**

Value: 1 up to 63

**Byte 8 up to 196: ItemVector**

Array (1.. NbrOfSupported SIDs) of Record (SupportedSID)

Record definition (up to 189 bytes):

**Byte 0-2: SID**

SID:

0x000000 - 0xFFFFFF

#### TP-LOG-TPL-REQ-195173/A-SID-9B-WifiHotspotMAC\_Rsp

Data Size: up to 40/22 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal Identifier**

0x9B: WifiHotspotMAC\_Rsp

**Byte 1: Utilization**

0x32: MobileCom\_Service2 - Embedded Modem

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result– Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 39/21 (Coding Table I / Coding Table II):**

MAC

Max. 18 characters, 17 plus 1 end of string

#### TP-LOG-TPL-REQ-207066/A-SID-9C-CCOISynchronizationSession\_Rq

Data Size: up to 82 bytes

**Byte 0: Signal Identifier**

0x9C: CCOISynchronizationSession\_Rq

**Byte 1: Utilization**

0x32: MobileCom\_Service2 - Embedded Modem

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result– Wait

**Byte 3: Character Coding**

Bit 0-5: Reserved

Bit 6-7: Coding

0x2: Coding Table III

0x00-0xFF RawData (Hexadecimal Notation)

**Byte 4-5: PolicyPlatformVersion**

**Byte 6-7: PolicyMajorVersion**

**Byte 8-9: PolicyMinorVersion**

**Byte 10-17: PolicyTableTimestamp**

**Byte 18-49: Hash Value of Policy Table Extension**

32 Bytes: SHA-256 value of unencrypted, uncompressed PolicyTableExtensionRawData

**Byte 50-81: Hash Value of User Friendly Messages**

32 Bytes: SHA-256 value of unencrypted, uncompressed UserFriendlyMessagesRawData

#### TP-LOG-TPL-REQ-207067/A-SID-9D-CCOISynchronizationSettings\_Rsp

Data Size: 17 up to 2061 bytes

**Byte 0: Signal Identifier**

0x9D: CCOISynchronizationSettings\_Rsp

**Byte 1: Utilization**

0x32: MobileCom\_Service2 - Embedded Modem

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result– Wait

**Byte 3: Character Coding**

Bit 0-5: Reserved

Bit 6-7: Coding

0x2: Coding Table III

0x00-0xFF RawData (Hexadecimal Notation)

**Byte 4-5: PolicyPlatformVersion**

**Byte 6-7: PolicyMajorRevision**

**Byte 8-9: PolicyMinorRevision**

**Byte 10-17: PolicyTableTimestamp**

**Byte 18-19: Length of Array**

0x0: Invalid

0x1: minimum length

0x3FF: maximum length

**Byte 20-21, 22-23, … Length of Array**

Bits 0-2: EntityType

0x0: tMeta

0x1: tData

0x2: tFunction

0x3: tFeature

0x4: tReserved1

…

0x7: tReserved4

Bits 3-9: EntityID

0x0: minimum

0x7F: maximum

Bit 10: bUAllowOnOff

Bit 11: bPAllowOnOff

Bit 12: bFPAllowOnOff

Bit 13: bSAllowOnOff

Bit 14: reserved

Bit 15: reserved

#### TP-LOG-TPL-REQ-207068/A-SID-9E-CCOISynchronizationAuthorizedUsers\_Rsp

Data Size: up to 4000 bytes

**Byte 0: Signal Identifier**

0x9E: CCOISynchronizationAuthorizedUsers\_Rsp

**Byte 1: Utilization**

0x32: MobileCom\_Service2 - Embedded Modem

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result– Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x2: Coding Table III

0x00-0xFF RawData (Hexadecimal Notation)

**Byte 4: Data Indication**

*Bit 0-5:* Reserved

*Bit 6:* 0x1: SyncP Encrypted, 0x0: Unencrypted

*Bit 7:* 0x1: GZIP Compressed, 0x0: Uncompressed

**Byte 5 up to 4005:**

Authorized Users Information

#### TP-LOG-TPL-REQ-207069/A-SID-9F-CCOISynchronizationSummaryReport

Data Size: 82 bytes

**Byte 0: Signal Identifier**

0x9F: CCOISynchronizationSummaryReport

**Byte 1: Utilization**

0x32: MobileCom\_Service2 - Embedded Modem

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result– Wait

**Byte 3: Character Coding**

Bit 0-5: Reserved

Bit 6-7: Coding

0x2: Coding Table III

0x00-0xFF RawData (Hexadecimal Notation)

**Byte 4-5: PolicyPlatformVersion**

**Byte 6-7: PolicyMajorVersion**

**Byte 8-9: PolicyMinorVersion**

**Byte 10-17: PolicyTableTimestamp**

**Byte 18-49: Hash Value of Policy Table Extension**

32 Bytes: SHA-256 value of unencrypted, uncompressed PolicyTableExtensionRawData

**Byte 50-81: Hash Value of User Friendly Messages**

32 Bytes: SHA-256 value of unencrypted, uncompressed UserFriendlyMessagesRawData

#### TP-LOG-TPL-REQ-207070/A-SID-A0-CCOISettingsUpdate\_Rq

Data Size: 17 up to 2061 bytes

**Byte 0: Signal Identifier**

0xA0: CCOISettingsUpdate\_Rq

**Byte 1: Utilization**

0x32: MobileCom\_Service2 - Embedded Modem

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result– Wait

**Byte 3: Character Coding**

Bit 0-5: Reserved

Bit 6-7: Coding

0x2: Coding Table III

0x00-0xFF RawData (Hexadecimal Notation)

**Byte 4-5: PolicyPlatformVersion**

**Byte 6-7: PolicyMajorRevision**

**Byte 8-9: PolicyMinorRevision**

**Byte 10-17: PolicyTableTimestamp**

**Byte 18-19: Length of Array**

0x0: Invalid

0x1: minimum length

0x3FF: maximum length

**Byte 20-21, 22-23, … Length of Array**

Bits 0-2: EntityType

0x0: tMeta

0x1: tData

0x2: tFunction

0x3: tFeature

0x4: tReserved1

…

0x7: tReserved4

Bits 3-9: EntityID

0x0: minimum

0x7F: maximum

Bit 10: bUAllowOnOff

Bit 11: reserved

Bit 12: reserved

Bit 13: reserved

Bit 14: reserved

Bit 15: reserved

#### TP-LOG-TPL-REQ-207875/A-SID-A1-SDARS\_ChannelList\_Rsp

Data size: up to 1057 (Coding Table II) bytes

**Byte 0: Signal identifier**

0xA1: SDARS\_ChannelList\_Rsp

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result – Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4: RspCode**

*Bit 0-7:RspCode*

0x0: Reserved

0x1: List Info

...

0x2..0xFF: Reserved

**Byte 5 - 6: NumberOfItemsTransmitted**

0x00: Invalid

0x01: 1

0x02: 2

....

0x1E: 30

0x1F – 0xFF: Reserved

**Byte 7 up to 1056 (Coding Table II): Channel List**

*Array(1..NumberOfItems) of record (ItemIndex, Channel Number, SID, Short Channel Name, Long Channel Name)*

*Record definition (up to 35 (Coding Table II) bytes):*

*Byte 0: ItemIndex*

0x00: Invalid

0x01: 1

0x02: 2

....

0x1E: 30

0x1F – 0xFF: Reserved

*Byte 1-2: Channel Number*

0x0000: 0

0x0001: 1

....

0x03E7: 999

0x3E8 – 0xFFFF: Reserved

*Byte 3-4: SID*

0x0000: 0

0x0001: 1

....

0x03E7: 999

0x3E8 – 0xFFFF: Reserved

*Byte 5 up to 34 (Coding Table II)*

*Short Channel Name*

Max. 8 characters plus 1 End Of String

*Long Channel Name*

Max. 20 characters plus 1 End Of String

***Note:*** *Both Long and Short channel names will be sent. The HMI will decide which to display.*

#### TP-LOG-TPL-REQ-208270/A-SID-A2-SDARS\_ChannelList\_Rq

Data size: up to 96 (Coding Table III) bytes

**Byte 0: Signal identifier**

0xA2: SDARS\_ChannelList\_Rq

**Byte 1: Utilization**

0x02: Radio\_Service2 – SDARS

**Byte 2: Command Execution Status**

0x00: Invalid/Inactive

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x2: Coding Table III

0x0000-0xFFFF Hexidecimal Notation

**Byte 4: OpCode**

*Bit 0-7:RspCode*

0x0: Reserved

0x1: Read

...

0x2..0xFF: Reserved

**Byte 5: NumberOfItemsRequested**

0x00: Invalid

0x01: 1

0x02: 2

....

0x1E: 30

0x1F - 0xFF: Reserved

**Byte 6 up to 95 (Coding Table III): Channel List Request**

*Array(1..NumberOfItems) of record (ItemIndex, SID)*

*Record definition (up to 3 (Coding Table III) bytes):*

*Byte 0: ItemIndex*

0x00: Invalid

0x01: 1

0x02: 2

....

0x1E: 30

0x1F - 0xFF: Reserved

*Byte 1-2: SID*

0x0000: 0

0x0001: 1

....

0x03E7: 999

0x3E8 – 0xFFFF: Reserved

#### TP-LOG-TPL-REQ-209648/A-SID-A3-MapVersionNumber\_St

Data Size: up to 68/36 (Coding Table I / Coding Table II) bytes

**Byte 0: Signal Identifier**

0xA3: MapVersionNumber\_St

**Byte 1: Utilization**

0x22: Nav\_Service2 - Navigation

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result– Wait

**Byte 3: Character Coding**

*Bit 0-5: Reserved*

*Bit 6-7: Coding*

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

0x1: Coding Table II

0x00-0xFF Latin-9 (1 byte per char)

**Byte 4 up to 67/35 (Coding Table I / Coding Table II):**

MapVersionNumber

Max. 32 characters, 31 plus 1 end of string

#### TP-LOG-TPL-REQ-211456/A-SID-A5-CCOIUserPrompt\_Rq

Data Size: up to 525 bytes

**Byte 0: Signal Identifier**

0xA5: UserPrompt\_Rq

**Byte 1: Utilization**

0x32: MobileCom\_Service2 - Embedded Modem

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result– Wait

**Byte 3: Character Coding**

Bit 0-5: Reserved

Bit 6-7: Coding

0x0: Coding Table I

0x0000-0xFFFF UNICODE UTF-16 (2 byte per char)

**Byte 4-7: User Prompt Request Token**

32-Bit Integer ID

**Byte 8: Prompt Type**

0x0: INVALID

0x1-0x40: Prompt Index

0x40-0xFF: Reserved

**Byte 9 up to 267: Variable Text1**

Variable text to display up to 128 characters + 1 end of string

**Byte 268 up to 524: Variable Text2**

Variable text to display up to 128 characters + 1 end of string

#### TP-LOG-TPL-REQ-211457/A-SID-A6-CCOIUserPrompt\_Rsp

Data Size: up to 9 bytes

**Byte 0: Signal Identifier**

0xA6: UserPrompt\_Rsp

**Byte 1: Utilization**

0x32: MobileCom\_Service2 - Embedded Modem

**Byte 2: Command Execution Status**

0x0y: Final Result – Success

0x1y: Final Result – Fail

0x2y: Final Result – Information

0x3y: Intermediate Result– Wait

**Byte 3: Character Coding**

Bit 0-5: Reserved

Bit 6-7: Coding

0x2: Coding Table III

0x00-0xFF RawData (Hexadecimal Notation)

**Byte 4-7: User Prompt Request Token**

*32-Bit Identifier*

**Byte 8:**

0x00: FAILED

0x01: TIMEOUT

0x02: SELECT\_NO

0x03: SELECT\_ASKMELATER

0x04: SELECT\_YES

0x05: SELECT\_YES\_PLUS\_OPTIONAL

0x06-0xFF: reserved

#### TP-LOG-TPL-REQ-201616/A-SID-CF-megaTP\_ConsecutivePackage

Data size: up to 5-4096 byte

**Byte 0: Signal identifier**

0xCF: megaTP\_ConsecutivePackageIndicator

**Byte 1: TotCPSet**

See megaTP specification

**Byte 2-3: mDataLength**

See megaTP specification

**Byte 4: mSID**

See megaTP specification

**Byte 5-4095: Data**

See megaTP specification

#### TP-LOG-TPL-REQ-201617/A-SID-FF-megaTP\_FirstPackage

Data size: up to 5-4096 byte

**Byte 0: Signal identifier**

0xFF: megaTP\_FirstPackageIndicator

**Byte 1: TotCPSet**

See megaTP specification

**Byte 2-3: mDataLength**

See megaTP specification

**Byte 4: mSID**

See megaTP specification

**Byte 5-4095: Data**

See megaTP specification

# Appendix: Reference Documents

|  |  |
| --- | --- |
| Reference # | Document Title |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |
| 9 |  |
| 10 |  |
| 11 |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |