Ultimate Remote Control

<<Feature>>

(F003071)

|  |  |  |  |
| --- | --- | --- | --- |
| Document Type | **Feature Implementation Specification (FIS)** | |  |
| Template Version | **6.1a** | |  |
| SysML Report Template Version | **O Beta (2021/03/10)** | |  |
| Document ID | **ffst01.10\_featuredocument\_sysmlreporttemplate** | |  |
| Document Location |  | |  |
| Document Owner | **Reed, Gregory (G.D.) (greed29)** | |  |
| Document Revision | **FIS Rev 2** | |  |
| Document Status | **Draft** | |  |
| Date Issued | **2021/05/18** | |  |
| Date Revised | **2021/05/18** | |  |
| Document Classification | GIS1 Item Number: | **27.60/35** |  |
| GIS2 Classification: | **Confidential** |

|  |  |  |  |
| --- | --- | --- | --- |
| Document Approval | | | |
| Person | Role | Email | Date |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

This document contains Ford Motor Company Confidential information. Disclosure of the information contained in any portion of this document is not permitted without the expressed, written consent of a duly authorized representative of Ford Motor Company, Dearborn, Michigan, U.S.A.

Copyright ©2021, Ford Motor Company

PRINTED COPIES ARE UNCONTROLLED

**Disclaimer**

**This document contains Ford Motor Company Confidential information. Disclosure of the information contained in any portion of this document is not permitted without the expressed, written consent of a duly authorized representative of Ford Motor Company, Dearborn, Michigan, U.S.A.**

This document contains information developed and accumulated by and for FORD MOTOR COMPANY. As such, it is a proprietary document, which, if disseminated to unauthorized persons, would provide others with restricted information, data, or procedures not otherwise available, exposing the FORD MOTOR COMPANY to potential harm.

Employees and suppliers having custody of this specification or authorized to use it must be cognizant of its proprietary nature and ensure that the information herein is not made available to unauthorized persons.

FORD MOTOR COMPANY reserves the right to protect this work as an unpublished copyrighted work in the event of an inadvertent or deliberate unauthorized publication. FORD MOTOR COMPANY also reserves its rights under copyright laws to protect this work as a published work.

This document or portions thereof shall not be distributed outside FORD MOTOR COMPANY without prior written consent. Refer all questions concerning disclosure to the author(s) or any duly authorized representative of Ford Motor Company.

**Copyright** © **2021 Ford Motor Company**

**Contents**

1 Introduction 6

1.1 Document Purpose 6

1.2 Document Scope 6

1.3 Document Audience 6

1.3.1 Stakeholder List 6

1.4 Document Organization 6

1.4.1 Document Context 6

1.4.2 Document Structure 6

1.5 Document Conventions 7

1.5.1 Requirements Templates 7

1.5.1.1 Identification of requirements 7

1.5.1.2 Requirement Attributes 7

2 Feature Implementation Overview 8

2.1 Description 8

2.2 Input Requirements 8

2.3 Assumptions 9

2.4 References 9

2.4.1 Ford Documents 9

2.4.2 External Documents and Publications 10

2.5 Glossary 10

2.5.1 Definitions 10

2.5.2 Abbreviations 11

3 Feature Implementation Architecture 12

3.1 Functional Architecture 12

3.1.1 Description 12

3.1.2 Function List 12

3.1.2.1 Functions of 13

3.1.2.2 Functions of 14

3.1.3 Signal List 14

3.2 Physical Architecture 18

3.2.1 E/E Architecture 18

3.2.1.1 E/E Architecture Variants 19

3.2.1.2 E/E Components 20

3.2.1.3 E/E Connections 20

3.2.1.4 Signal List 22

3.2.2 Software Component Architecture 22

3.2.2.1 Description 22

3.3 Function Deployment 22

3.3.1 Deployment Variants 23

3.3.1.1 Main (Only) variant 23

3.3.1.2 Deployment “Variant 1” 23

3.3.2 Function Allocation 24

3.3.2.1 Functional Safety 25

4 Feature Implementation Modeling 27

4.1 Component Interaction Diagrams 27

4.1.1 Scenario: “System Startup / Shutdown” 27

4.1.2 Scenario: “Normal Operation” 27

4.1.3 Functional Safety 28

4.1.3.1 Fault Handling Time Analysis 28

4.1.3.2 Requirements Derivation Diagram 29

4.2 Component Interface Behavior Diagrams 29

5 Feature Implementation Requirements 30

6 Open Concerns 31

7 Revision History 32

8 Appendix 33

8.1 Data Dictionary 33

8.1.1 Logical Signals 33

8.1.2 Logical Parameters 33

8.1.3 Technical Signals 33

8.1.3.1 GSDB Signals 34

8.1.3.2 Service Oriented Communication 34

8.1.3.3 Hardwired Signals 34

8.1.3.4 Diagnostic Interfaces 34

8.1.4 Technical Parameters 36

8.1.5 Mappings 36

8.1.6 Technical Interfaces 37

8.1.6.1 AIS Interfaces 37

8.1.6.2 Service Oriented Communcation (SoC) Interfaces 38

8.1.6.3 AUTOSAR Ports (SW Interfaces) 38

8.1.7 Messages 38

8.1.7.1 CAN Bus “<Bus Name>” 38

8.1.7.2 LIN Bus “<Bus Name>” 39

8.1.8 Encoding Types 39

**List of Figures**

Figure 1: 21

Figure 2:  24

Figure 3: 30

**List of Tables**

Table 1‑1: Electrical Architecture(s) referenced in this document 6

Table 1‑2: Functions referenced in this document 6

Table 2‑1: Ford Internal Documents 10

Table 2‑2: Ford Internal Documents *(not specified in SysML model)* 11

Table 2‑3: External Documents and Publications 11

Table 2‑4: External Documents and Publications *(not specified in SysML model)* 11

Table 2‑5: Definitions Used In This Document 12

Table 2‑6: Abbreviations used in this document 12

Table 3‑1: List of Functions 15

Table 3‑2: List of ‑‑ Functions 15

Table 3‑3 List of signals sent by 18

Table 3‑4 List of signals received by 19

Table 3‑5: Electrical Components 21

Table 3‑6: E/E Connections for 23

Table 3‑7: Function Allocation Table 26

Table 3‑8: Function Allocation Table 27

Table 3‑9: Architectural Redundancy Summary 27

Table 4‑1: Fault Handling Time Table 30

Table 5‑1: Input Signal mappings of ‑ 34

Table 5‑2: Output Signal mappings of ‑ 35

Table 5‑3: Parameter mappings of ‑ 36

Table 5‑4: Component Specific Requirements 37

Table 5‑5: Inherited Requirements 37

Table 5‑6: Input Signal mappings of ‑ 39

Table 5‑7: Output Signal mappings of ‑ 40

Table 5‑8: Parameter mappings of ‑ 40

Table 5‑9: Component Specific Requirements 40

Table 5‑10: Inherited Requirements 40

Table 5‑11: Input Signal mappings of Component: ‑ 41

Table 5‑12: Output Signal mappings of Component: ‑ 42

# Introduction

## Document Purpose

The Feature Implementation Specification (FIS) specifies the deployment of the logical functions of a feature to an electrical architecture. The FIS specifies all interactions between the ECUs of the electrical architecture required for the feature including the technical signals and the interfaces. It also gives interface and integration requirements, which are specific to the feature for the electrical architecture.

To get more information about the concept of feature, function and component level abstraction refer to the [Ford RE Wiki](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Engineering+for+SW+Enabled+Features).

## Document Scope

This FIS describes the deployment of the feature <Ultimate Remote Control > to the following electrical architecture(s):

FNV3

## Document Audience

The FIS is authored by URC Team - . All Stakeholders, i.e., all people who have a valid interest in the feature implementation should read and, if possible, review the FIS. It needs to be guaranteed, that all stakeholders have access to the currently valid version of the FIS.

### Stakeholder List

For the latest list of the function stakeholders and their roles & responsibilities refer to <<https://www.vsemweb.ford.com/tc/launchapp?-attach=true&-s=226TCSession&-o=C$pBjd1Gx3NrTDAAAAAAAAAAAAA&servername=Production_Server>>.

## Document Organization

### Document Context

Refer to the [Specification Structure page](http://wiki.ford.com/display/RequirementsEngineering/Specification+templates) in the [Ford RE Wiki](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Engineering+for+SW+Enabled+Features) to understand how the FIS relates to other Ford Requirements Documents and Specifications.

### Document Structure

The structure of this document is explained below:

**Section 1** – Introduction – Giving an explanation how to use this document including responsibilities and the scope of the document. Additionally it contains the revision history and a list of unsettled but known issues that have to be consolidated in future versions. It explains the terminology and gives a clarification of the definitions, concepts and abbreviations used in the document.

**Section 2** – Feature Implementation Description – Giving an overview of the platform and listing assumptions, constraints or dependencies

**Section 3** – Feature Implementation Architecture – Describing 3 Architecture Views:

* Functional Architecture – Showing the logical architecture of functions
* Physical Architecture – Showing the physical architecture (first of all the E/E Architecture), which the Logical Functions get allocated to.
* Software Architecture – Showing the software architecture relevant for the feature (for features with in-house development only)
* Function Deployment – Presenting the allocation of logical functions and signals to the electrical and other components

**Section 4** – Deployment Specific Modeling –Modeling techniques providing additional detail on e.g. interface behavior

**Section 5** – Deployment Specific Requirements – Deployment specific requirements for ECUs, Network Communication, and Process

**Section 6** – List of Open Concerns

**Section 7** – Revision History

**Section 8** – Appendix - Presenting additional data mainly in a tabular form, e.g., a data dictionary

## Document Conventions

### Requirements Templates

Refer to “[How to use the Specification Templates](http://wiki.ford.com/display/RequirementsEngineering/How+to+use+the+Specification+Templates?src=contextnavpagetreemode)” on how to use the specification templates and the VBA macros to create/edit the requirements in the specifications.

The VBA macro enable the import of the specification to VSEM (refer to ["How to import specifications into VSEM as separate requirements"](http://wiki.ford.com/pages/viewpage.action?pageId=104991616&src=contextnavpagetreemode)).

#### Identification of requirements

The unique requirement ID given in the headline of any requirement follows the requirement throughout the development process. The requirement ID format follows a well-defined syntax.

All identifiers in an FIS shall be composed of 4 parts:

* A leading prefix, which indicates the type of requirement (R=Requirement, UC=Use Case, SC=Scenario, …)
* A prefix, which indicates the abstraction level (F=Feature, FNC=Function, CMP = component).
* Followed by a name, indicating the scope, which the requirement belongs to (e.g. feature or function name )
* Ending with the actual requirement number

*Example:*

*R\_CMP\_LockArbitrator\_00004* This is the fourth requirement on component level for the function Lock Arbitrator.

#### Requirements Attributes

Additionally attributes can be added to each requirement. This helps to classify requirements. A [list of available attributes](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes?src=contextnavpagetreemode) is given in the RE Wiki.

## References

### Ford Documents

The list of all Ford internal documents, which are directly related.

See Section 2.2 for all Ford Input and reference documents

### External Documents and Publications

The list of external documents could include books, reports and online sources.

| **Reference** | **Document / Publication** |
| --- | --- |
| IEEE Std 1012-2004 IEEE Standard for Software Verification and Validation |  |
|  |  |

Table 1‑3: External documents and publications

## Glossary

### Definitions

| **Definition** | **Description** |
| --- | --- |
| Climate Zone | Number of Climate Zones defined by Climate System (One Zone, Dual Zone, Tri Zone, Quad Zone) URC user’s control the climate zone they are occupy |
| Digital Shade Feature | Digital Shade is a feature that allows different sections of the sunroof to be shaded by the User |
| Permanent URC User | Re-occurring use of URC in a vehicle. User’s phone ID is stored in vehicle infotainment system in a permanent connection list until manual deletion or master reset. Example: Family vehicle |
| Rear Seat Controls Lockout Feature | Feature that provides a way for the Driver/1R passenger to disable rear seat occupants from controlling vehicle controls (Climate, Audio, URC) |
| Rejuvenate Feature | Feature that provides a “Rejuvenate” experience for the driver when the vehicle is stationary – Disables parts of URC |
| Seat Climate | Seat heating or venting of the seat surface |
| Seat Massage | Seat Massage feature allows the control of massaging seats (Multi-Contour Seats) by inflating and deflating air bladders in the cushion and back of the seat |
| Seat Position Settings | The Seat settings consists of  a. **Seat Cusion Movement**  1. Cushion track fore/aft  2. Cushion track up/down  3. Cushion track tilt  4. Cushion extension left  5. Cushion extension right  6. Calf Raise  b. **Seat Back Movement**  1. Back recline  2. Back upper pivot  3. Lumbar (Power Mechanical)  c. **Head Rest Movement**  1. Head Rest fore/aft  2. Head Rest up/down  d. **Bladders**  1. Back Bolster bladders  2. Cushion Bolster bladders  3. Lumbar (Bladders) |
| Seating Zone/Seating Area | A seating zone is defined by each outboard seat per row (left and right). Example: 6 seating zones for a 3 row vehicle |
| Stationary | Transmission in Park |
| Temporary URC User | Once time use of URC in a vehicle. User’s phone ID is not stored in vehicle infotainment system. Example: Riding to lunch or ride share. |
| URC User | Vehicle Occupants intending to use URC Hand Held Device App |
| Vehicle Occupant | Second or Third Row Passengers |
| Zone Audio Settings | Zone Audio adds Speakers to each outboard seat allowing the User individual audio functions (My Seat Space Feature):  a. In car communication ICC (Seat to Seat Communication)  b. Independent sound zones (My Seat Space, Individual Music Streaming)  c. Private Phone Call Localization  d. Shared music across zones (Media Sharing) |

Table 1‑4: Definitions used in this document

### Abbreviations

| **Abbr.** | **Stands for** |
| --- | --- |
| 1R | First Row Occupant |
| 2R | Second Row Occupant |
| 3R | Third Row Occupant |
| ARL | Attribute Release Letter |
| A Status | Audio Status |
| C Status | Climate Status |
| L Status | Lighting Status |
| HHD | Hand Held Device |
| HMI | Human Machine Interface |
| MSS | My Seat Space |
| MVP | Minimal Viable Product |
| RSCL | Rear Seat Controls Lockout Feature |
| URC | Ultimate Remote Control |

Table 1‑5: Abbreviations used in this document.

# Feature Implementation Overview

## Description

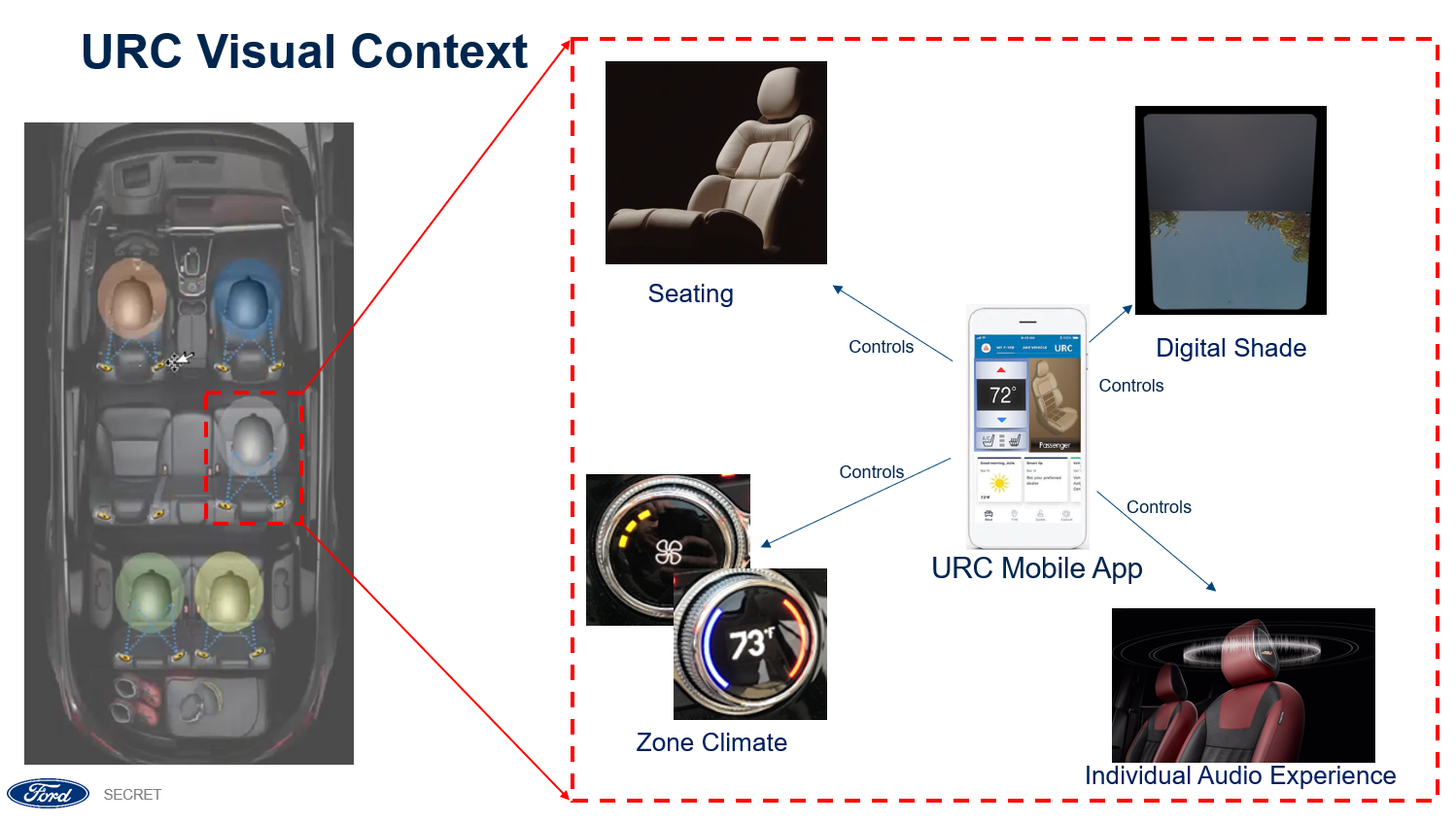
F003071 Ultimate Remote Control

URC Feature allows the passengers control of their environment inside the vehicle for each individual area or seating zone.

It is simple and centralized way for first row, second, and third row passenger(s) to control their own seat functions (position, climate, massage), zone climate adjustments, and sound/audio with a Hand Held Device (HHD).

\* The HHD will connect with the vehicle with BLE

\* The URC User(s) will select which seat/area zone they are sitting in and take control for their seating zone



## Input Requirements/Documents

|  |  |  |  |
| --- | --- | --- | --- |
| **Reference**  (Reference as listed in ch. “References”) | **Section/Requirement** | **Description** | **Derived Requirement**  (optional – reference to requirement in ch. “Feature Implementation Requirements”) |
| **Feature/Function Requirements** | | | |
| VDOC080630 | URC Feature Document | URC specification feeding FIS document |  |
| VDOC080631 | URC Function Specification | URC specification feeding FIS document |  |
| VSEM ID: 735414 | Seat Service Specification | Describes the Seat Service command and control with URC as an input |  |
| VSEM ID: 814606-A-004 | Climate System Interface Spec | Describes the signals URC will use for climate control |  |
| **Ford Engineering Standards** | | | |
|  |  |  |  |
|  |  |  |  |
| **Legal Regulations** | | | |
|  | Compliance with FMVSS101 | The Feature shall comply with FMVSS101. |  |
|  |  |  |  |
| **Industry Standards** | | | |
|  | ISO 26262 | The system should be developed according to Ford's implementation of Functional Safety. |  |
|  |  |  |  |
| **Other Sources** | | | |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table 6: Input Requirements/Documents

## Lessons Learned

See URC FMA Quality History in FEDE for Lessons Learned

## Assumptions

See URC Feature doc for high level assumptions

For Implementation level assumptions:

ASMP-01 URC CAN Control

In order for URC to control a component, the component must have signals on the network and a message pathway to the infotainment system. URC should not have to create any “new” signals because it is controlling existing systems

|  |  |
| --- | --- |
| **Purpose** | **Assumption Category** |
| URC Feature should not be intrusive to the vehicle network |  |

ASMP-02 URC and SoA

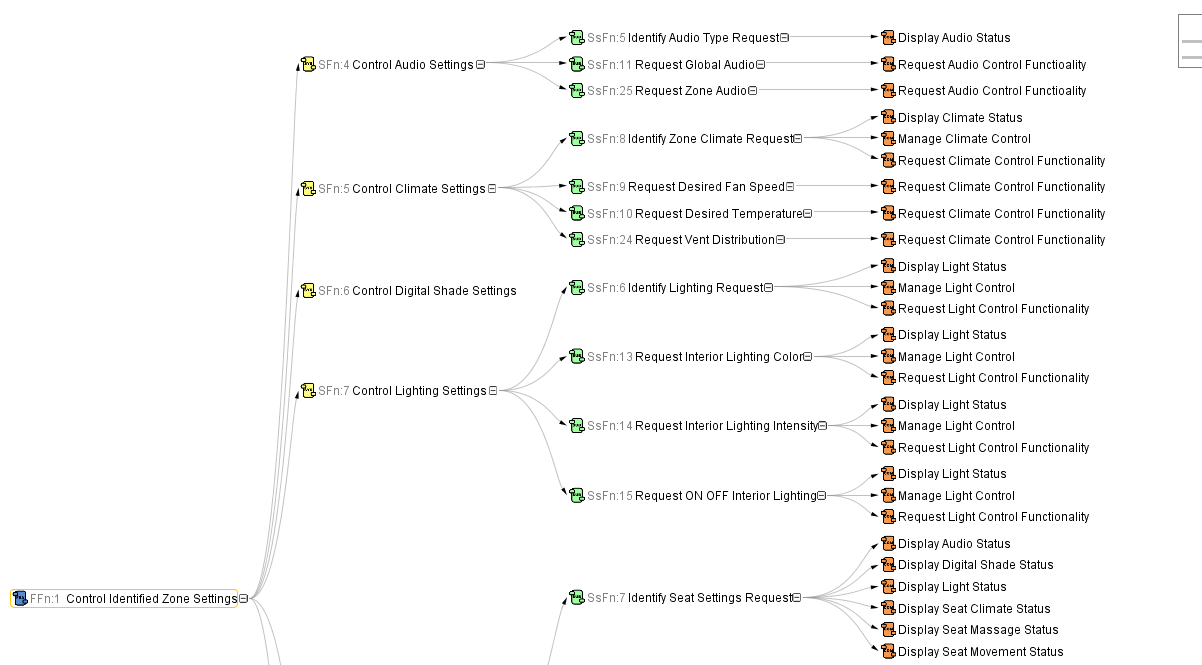
URC shall utilize the Seat Service for command and controlling of Seat position and Seat massage signals

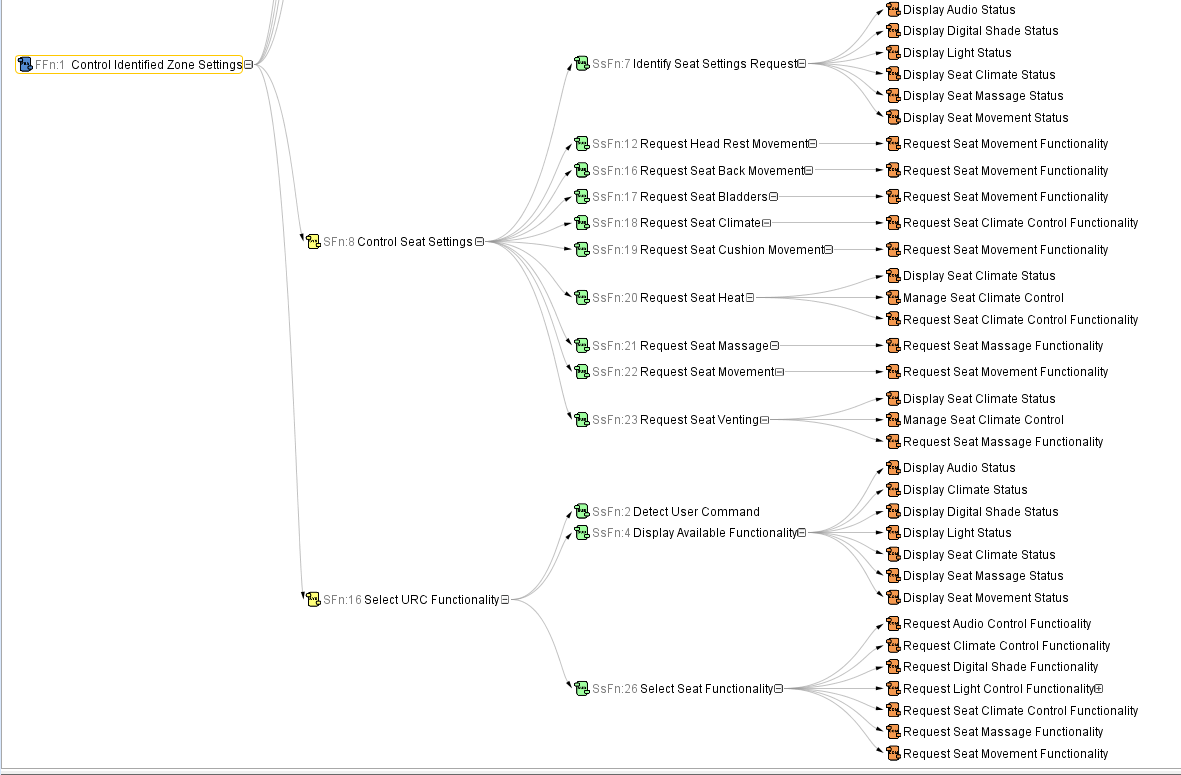
|  |  |
| --- | --- |
| **Purpose** | **Assumption Category** |
| Utilize existing services |  |

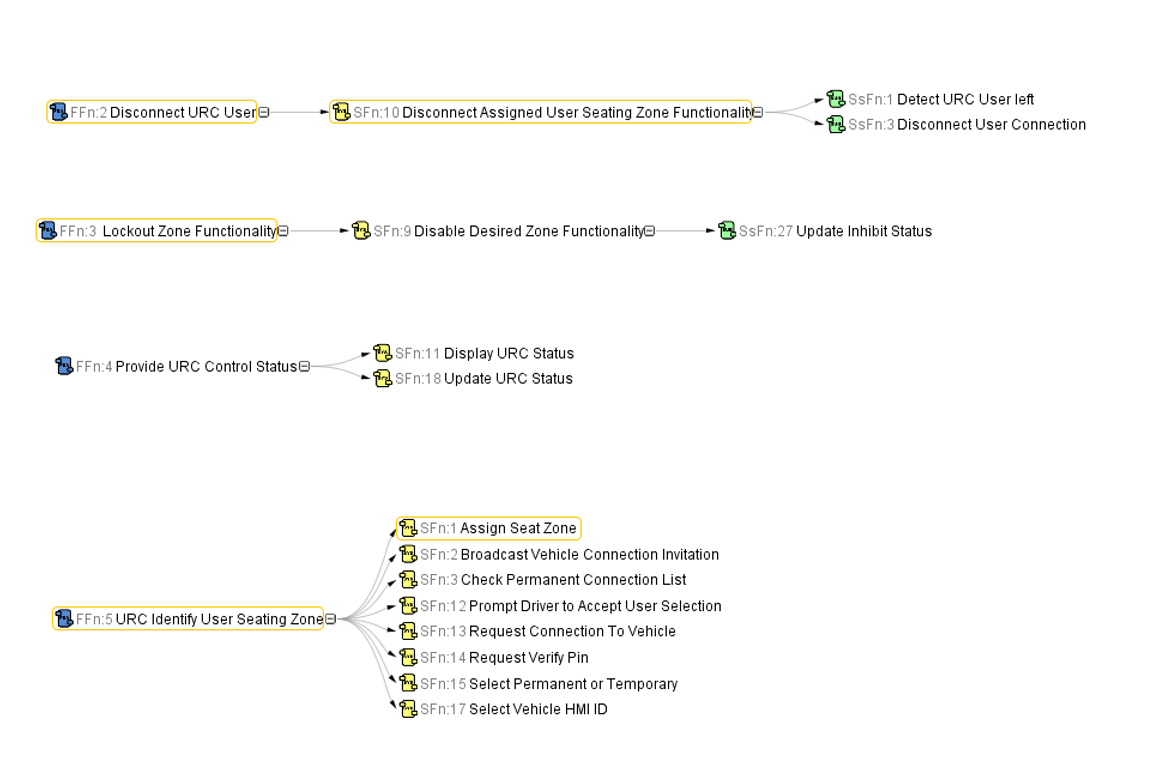
# Feature Implementation Architecture

## Functional Architecture

### Functional Decomposition







### Function List

The following functions from the [Global Feature & Function List](https://www.vsemweb.ford.com:443/tc/launchapp?-attach=true&-s=226TCSession&-o=ZmZNi0JHx3NrTDAAAAAAAAAAAAA) are referenced in this Feature Implementation Specification:

Implementation Function List:

|  |  |
| --- | --- |
| IFn:1 | Display Audio Status |
| IFn:2 | Display Climate Status |
| IFn:3 | Display Digital Shade Status |
| IFn:4 | Display Seat Climate Status |
| IFn:5 | Display Seat Massage Status |
| IFn:6 | Display Seat Movement Status |
| IFn:7 | Manage Audio Control |
| IFn:8 | Manage Climate Control |
| IFn:9 | Manage Digital Shade Control |
| IFn:10 | Manage Seat Climate Control |
| IFn:11 | Manage Seat Massage Control |
| IFn:12 | Manage Seat Movement Control |
| IFn:13 | Request Audio Control Functionality |
| IFn:14 | Request Climate Control Functionality |
| IFn:15 | Request Digital Shade Functionality |
| IFn:16 | Request Seat Climate Control Functionality |
| IFn:17 | Request Seat Massage Functionality |
| IFn:18 | Request Seat Movement Functionality |
| IFn:19 | Manage CAN Data |
| IFn:20 | Manage Lockout URC |
| IFn:21 | Manage Seat Selection |
| IFn:22 | Manage URC User Connections |
| IFn:23 | Manage Zone Database |

### Signal List

See section 5.2 for signal list for each module

## Physical Architecture

### E/E Architecture

#### E/E Architecture Variants

**FNV3 Architecture with Services**

#### E/E Components

|  |  |
| --- | --- |
| Component Name | **Description** |
| ACM (AHU) | Audio Control Module |
| APIM | Accessory Protocol Interface Module |
| ECG | Enhanced Central Gateway |
| SCMC (MCSM) | Seat Ctrl Mod C (MultiContour Seat - Rr Drv Side) |
| SCMD (MCSM) | Seat Ctrl Mod D (MultiContour Seat - Rr Pass Side) |
| SCMH (MCSM) | Seat Ctrl Mod H (MultiContour Seat - Frt Pass Side) |
| SCMK | Seat Ctrl Mod K (2nd Row Seat - Left Rear) |
| SCML | Seat Ctrl Mod L (2nd Row Seat - Right Rear) |
| SCMB\_PSM | Seat Ctrl Mod B (Pass. Seat Module) |
| RCCM | Remote Climate Control Module |

Table 3‑2: Electrical Components

#### E/E Connections

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Connection Name | **Connection Type** | **Protocol**  Only if ‘Connection Type’ is “Network”/”RF-Digital” | **Description** | **Allocated Messages**  Only if ‘Connection Type’ is “Network”/”RF-Digital” | **Connected Nodes** |
| FD1-CAN | Network | CAN FD | CAN Flexible Data Enhancement for module to module communication |  |  |
| FD3-CAN | Network | CAN FD | CAN Flexible Data Enhancement for module to module communication |  |  |
| HS3-CAN | Network | CAN (High Speed) | High Speed Infotainment CAN for module to module communication |  |  |
| MS1-CAN | Network | CAN (Mid Speed) | Medium Speed CAN for module to module communication |  |  |
| Ethernet | Network | Ethernet (MQTT) | Module to module communication (mainly ECG to APIM) |  |  |

#### Signal List

See section 5.2 for signal list for each module

### Software Component Architecture

#### Description

## Function Deployment

### Deployment Variants

URC will utilize a function call Zone Database to relate the signals in APIM to the Hand Held Device. This populates the Hand Held Device HMI with the controls with respect to the seat selection.

### Function Allocation

|  |  |  |
| --- | --- | --- |
| Component | Technology Function Name | Logical Function Name |
| Hand Held Device | Display Audio Status | See section 5.2 for signal list per module |
| Display Climate Status | See section 5.2 for signal list per module |
| Display Digital Shade Status | See section 5.2 for signal list per module |
| Display Seat Climate Status | See section 5.2 for signal list per module |
| Display Seat Massage Status | See section 5.2 for signal list per module |
| Display Seat Movement Status | See section 5.2 for signal list per module |
| Request Climate Control Functionality | See section 5.2 for signal list per module |
| Request Seat Climate Functionality | See section 5.2 for signal list per module |
| Request Digital Shade Functionality | See section 5.2 for signal list per module |
| Request Seat Massage Functionality - 2R Drv Side | See section 5.2 for signal list per module |
| Request Seat Massage Functionality - 2R Pass Side | See section 5.2 for signal list per module |
| Request Seat Massage Functionality - 1R Pass Side | See section 5.2 for signal list per module |
| Request Seat Movement Functionality - 2R Drv Side | See section 5.2 for signal list per module |
| Request Seat Movement Functionality - 2R Pass Side | See section 5.2 for signal list per module |
| Request Seat Movement Functionality - 1R Pass Side | See section 5.2 for signal list per module |
| Request Audio Control | See section 5.2 for signal list per module |
| APIM | Manage Lockout URC | See section 5.2 for signal list per module |
| Manage Seat Selection | See section 5.2 for signal list per module |
| Manage URC User Connections | See section 5.2 for signal list per module |
| Manage Zone Database | See section 5.2 for signal list per module |
| AHU | Manage Audio Control | See section 5.2 for signal list per module |
| RCCM | Manage Climate Control | See section 5.2 for signal list per module |
| Manage Seat Climate | See section 5.2 for signal list per module |
| OZM | Manage Digital Shade | See section 5.2 for signal list per module |
| SCMC | Manage Seat Massage - 2R Drv Side | See section 5.2 for signal list per module |
| SCMD | Manage Seat Massage - 2R Pass Side | See section 5.2 for signal list per module |
| SCMH | Manage Seat Massage - 1R Pass Side | See section 5.2 for signal list per module |
| SCMK | Manage Seat Movement - 2R Drv Side | See section 5.2 for signal list per module |
| SCML | Manage Seat Movement - 2R Pass Side | See section 5.2 for signal list per module |
| SCMB\_PSM | Manage Seat Movement - 1R Pass Side | See section 5.2 for signal list per module |
| ECG | Manage CAN Data | See section 5.2 for signal list per module |

Table 3‑5: Function Allocation Table (Basic)

# Feature Implementation Modeling

## Component Interaction Diagrams

### Scenario: “System Startup / Shutdown”

### Scenario: “Normal Operation”

No “Feature Scenario Diagram” found.

## Component Interface Behavior Diagrams

*Not supported by MagicDraw report generation.*

# Feature Implementation Requirements

## Functional Safety

### ASIL Decomposition of Technical Safety Requirements

<Place the input TSR here above the decomposition table>

| **Input TSR** | <Provide the ID of the TSR which shall be decomposed. That TSR is given above> | |
| --- | --- | --- |
| **Decomposition Rationale** | <Give a reason why the decomposition was performed> | |
| **Method for Decomposition** | Choose a Method | |
| **TSR 1 after Decomposition** | **TSR ID** | <Provide the ID of the decomposed TSR> |
| **TSR Title** | <Provide the title of the decomposed TSR> |
| **ASIL** |  |
| **Rationale** | <Provide a reason and thought behind that requirement. Should include how the requirement is able to independently fulfill the needs of the parent requirement> |
| **Satisfied by** | <Provide a Technology Function, physical signal, or physical component satisfying the requirement. This element shall be independent of the element satisfied by the other half of the ASIL decomposition.> |
| **TSR 2 after Decomposition** | **TSR ID** | <Provide the ID of the decomposed TSR> |
| **TSR Title** | <Provide the title of the decomposed TSR> |
| **ASIL** |  |
| **Rationale** | <Provide a reason and thought behind that particular requirement. Should include how the requirement is able to independently fulfill the needs of the parent requirement> |
| **Satisfied by** | <Provide a Technology Function, physical signal, or physical component satisfying the requirement. This element shall be independent of the element satisfied by the other half of the ASIL decomposition.> |
| **TSR for Independence**  *Note: should consider commonly used input, output and processing*  *Note: additional row should be added if additional* *requirements for Independence are necessary* | **TSR ID** |  |
| **TSR Title** |  |
| **ASIL** |  |
| **Rationale** |  |

Table 5‑1: ASIL Decomposition Table

## Requirements on Components

### Hand Held Device

Hand Held Device

#### Technology Function 1827883074.jpg **Display Audio Status**

##### Function Interfaces

This Function displays the current status of Audio on the URC User’s HHD app

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See AHU section for Display Audio Status. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See AHU section for Display Audio Status. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function 1435183513.jpg **Display Climate Status**

##### Function Interfaces

This Function displays the current status of climate on the URC User’s HHD app

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | | **Technical Signal Name** | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See RCCM section for Display Climate Status. | |  | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | 6.0 | | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | | **Technical Signal Name** | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See RCCM section for Display Climate Status. | |  | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | 6.0 | | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function -1360470560.jpg **Display Digital Shade Status**

##### Function Interfaces

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See OZM section for Display Digial Shade Status. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See OZM section for Display Digial Shade Status. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function -658924689.jpg **Display Seat Climate Status**

##### Function Interfaces

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See RCCM section for Display Seat Climate Status. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See RCCM section for Display Seat Climate Status. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
| REQ\_abc |  | Removed | -- |  |
| REQ\_def |  | Replaced | REQ\_xyz |  |
| -- |  | Added | REQ\_123 |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function 1578538233.jpg **Display Seat Massage Status**

##### Function Interfaces

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See SCMC, SCMD, SCMH sections for Display Seat Massage Status. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See SCMC, SCMD, SCMH sections for Display Seat Massage Status. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function -994081234.jpg **Display Seat Movement Status**

##### Function Interfaces

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See SCMK, SCML, SCMB\_PSM sections for Display Seat Movement Status. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See SCMK, SCML, SCMB\_PSM sections for Display Seat Movement Status. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function -1650879436.jpg **Request Climate Control Functionality**

##### Function Interfaces

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See RCCM section for Request Climate Control Functionarlity. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See RCCM section for Request Climate Control Functionarlity. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function 950530537.jpg **Request Seat Climate Control Functionality**

##### Function Interfaces

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See RCCM section for Request Seat Climate Control Functionarlity. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See RCCM section for Request Seat Climate Control Functionarlity. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function 449209574.jpg **Request Digital Shade Functionality**

##### Function Interfaces

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See OZM section for Request Digital Shade Functionarlity. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See OZM section for Request Digital Shade Functionarlity. |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function -347357886.jpg **Request Seat Massage Functionality**

##### Function Interfaces

The URC User’s Hand Held devices initiates the request and then displays the status.

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See SCMC, SCMD, SCMH sections for Request Seat Massage Functionality |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See SCMC, SCMD, SCMH sections for Request Seat Massage Functionality |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function -446258746.jpg **Request Seat Movement Functionality**

##### Function Interfaces

###### Inputs

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See SCMK, SCML, SCMB\_PSM sections for Request Seat Movement Functionality |  | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See SCMK, SCML, SCMB\_PSM sections for Request Seat Movement Functionality |  | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function -262039083.jpg **Request Audio Control Functionality**

##### Function Interfaces

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See AHU section for Request Audio Control Functionality |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| See AHU section for Request Audio Control Functionality |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

### APIM

#### Technology Function -85992038.jpg **Manage Lockout URC**

##### Function Interfaces

The Rear Seat Controls Lockout Feature shall provide the lockout functionality of URC

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| URC Feature enable | HMI\_HMIMode\_St = ON | | |  |  |  |
| URC Feature disable | HMI\_HMIMode\_St = OFF | | |  |  |  |
|  |  | | |  |  |  |
| Lockout URC 2R/3R | TBD | | |  | Internal to APIM |  |
| Unlock URC 2R/3R | TBD | | |  | Internal to APIM |  |
|  |  | | |  |  |  |
| Lockout Climate URC 2R/3R | TBD | | |  | Internal to APIM |  |
| Unlock Climate URC 2R/3R | TBD | | |  | Internal to APIM |  |
|  |  | | |  |  |  |
| Lockout Audio URC 2R/3R | TBD | | |  | Internal to APIM |  |
| Unlock Audio URC 2R/3R | TBD | | |  | Internal to APIM |  |
| URCL\_HMI\_rq | TBD | | |  | Internal to APIM |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

Table 5‑2: Input Signal mappings of Function Manage Control Request

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| URC Feature enable | HMI\_HMIMode\_St = ON | | |  |  |  |
| URC Feature disable | HMI\_HMIMode\_St = OFF | | |  |  |  |
|  |  | | |  |  |  |
| Lockout URC 2R/3R Status | TBD | | |  | Internal to APIM |  |
| Unlock URC 2R/3R Status | TBD | | |  | Internal to APIM |  |
|  |  | | |  |  |  |
| Lockout Climate URC 2R/3R Status | TBD | | |  | Internal to APIM |  |
| Unlock Climate URC 2R/3R Status | TBD | | |  | Internal to APIM |  |
|  |  | | |  |  |  |
| Lockout Audio URC 2R/3R Status | TBD | | |  | Internal to APIM |  |
| Unlock Audio URC 2R/3R Status | TBD | | |  | Internal to APIM |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

Table 5‑3: Output Signal mappings of Function Manage Control Request

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function 1767413114.jpg **Manage Seat Selection**

##### Function Interfaces

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| URC User select Seat | PASSENGER | | |  | BLE |  |
| URC User select Seat | SECOND\_ROW\_LH | | |  | BLE |  |
| URC User select Seat | SECOND\_ROW\_RH | | |  | BLE |  |
| URC User select Seat | THIRD\_ROW\_LH | | |  | BLE |  |
| URC User select Seat | THIRD\_ROW\_LH | | |  | BLE |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

Table 5‑2: Input Signal mappings of Function Manage Climate Control

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Publisher Interface** | **Connection**  (*Optional)* |
| Output of Seat Selection goes to Zone Database – See Zone database section |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

Table 5‑3: Output Signal mappings of Function Manage Climate Control

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function 1912288002.jpg **Manage URC User Connections**

##### Function Interfaces

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| URC Feature enable | HMI\_HMIMode\_St = ON | | |  |  |  |
| URC User request connection | TBD | | |  | BLE |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

Table 5‑2: Input Signal mappings of Function Manage Seat Massage Control

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| URC User Connection Successful | URC User Connection Successful | | |  | BLE |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

Table 5‑3: Output Signal mappings of Function Manage Seat Massage Control

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function 1912288002.jpg **Manage Zone Database**

##### Function Interfaces

Zone Database will receive the status via CAN or SoA from the systems available to control for URC Users per seating zone. This data populates the URC User’s hand held device with the correct HMI Controls

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Receive Zone Fucntionality based on seat selection | TBD | | |  | BLE |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

Table 5‑2: Input Signal mappings of Function Manage Seat Massage Control

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Send zone content to Hand held device | TBD | | |  | BLE |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

Table 5‑3: Output Signal mappings of Function Manage Seat Massage Control

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

### AHU

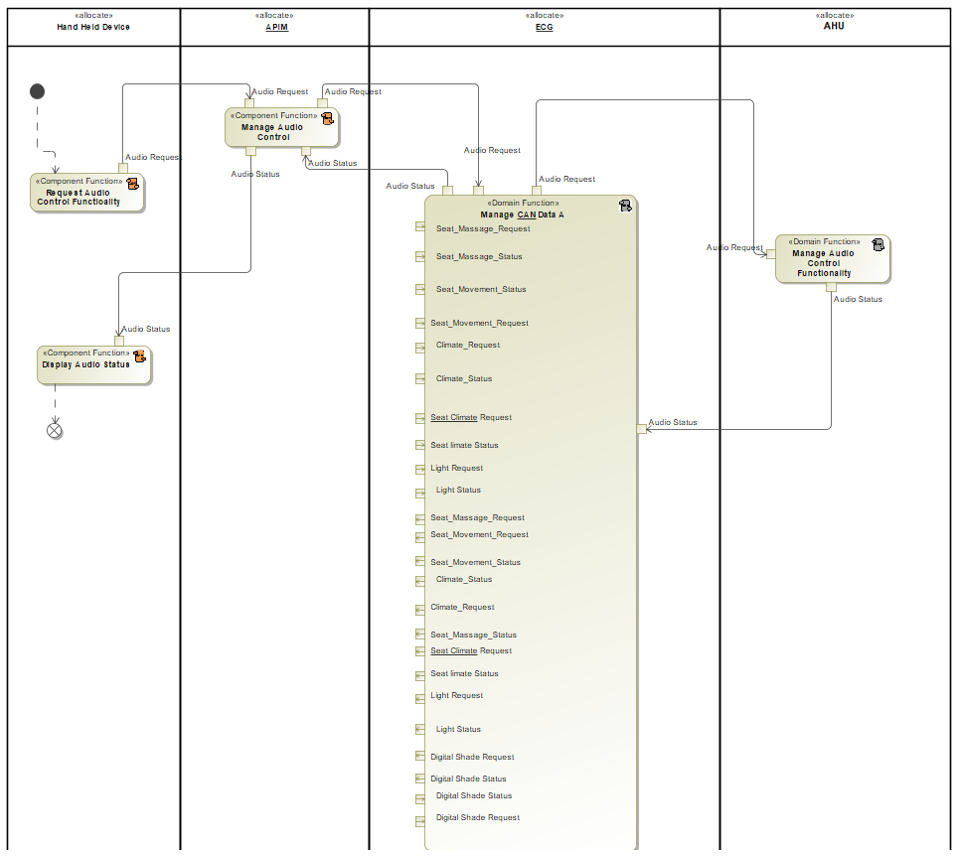
Audio Modules AHU

#### Technology Function 1912288002.jpg **Manage Audio Control**

##### Function Interfaces

See My Seat Space FIS for detailed Zone Audio Signals

**Audio Control Behavior:**



URC User’s shall be able to share an audio source to the vehicle audio system

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Request Global Audio Control | Request\_Share\_Own\_Source\_Full\_Cabin | | |  | Placeholder |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| Request Zone Audio Control | Request\_Audio\_Share | | |  | Placeholder |  |
| Request Zone Audio Control | Request\_ICC\_On | | |  | Placeholder |  |
| Request Zone Audio Control | Request\_ICC\_Off | | |  | Placeholder |  |
| Request Zone Audio Control | Request\_Zone\_Volume\_Inc | | |  | Placeholder |  |
| Request Zone Audio Control | Request\_Zone\_Volume\_Dec | | |  | Placeholder |  |
| Request Zone Audio Control | Request\_DnD\_On | | |  | Placeholder |  |
| Request Zone Audio Control | Request\_DnD\_Off | | |  | Placeholder |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

Table 5‑2: Input Signal mappings of Function Manage Control Request

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Display Global Audio status | Play\_Share\_Own\_Source\_Full\_Cabin | | |  | Placeholder |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| Display Zone Audio status | Audio\_Share\_Status | | |  | Placeholder |  |
| Display Zone Audio status | Request\_ICC\_Status | | |  | Placeholder |  |
| Display Zone Audio status | Request\_Zone\_Volume\_status | | |  | Placeholder |  |
| Display Zone Audio status | Request\_DnD\_Status | | |  | Placeholder |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

Table 5‑3: Output Signal mappings of Function Manage Control Request

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
| REQ\_abc |  | Removed | -- |  |
| REQ\_def |  | Replaced | REQ\_xyz |  |
| -- |  | Added | REQ\_123 |  |

Table 5‑5: Component Specific Requirements

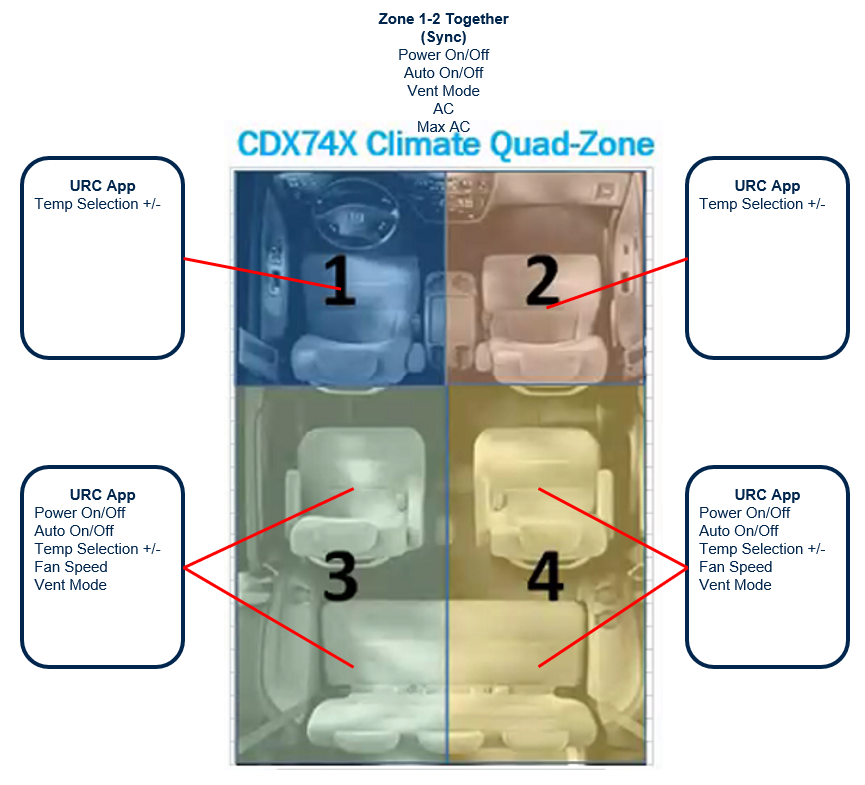
|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

### RCCM

#### Technology Function -340505036.jpg **Manage Climate Control**

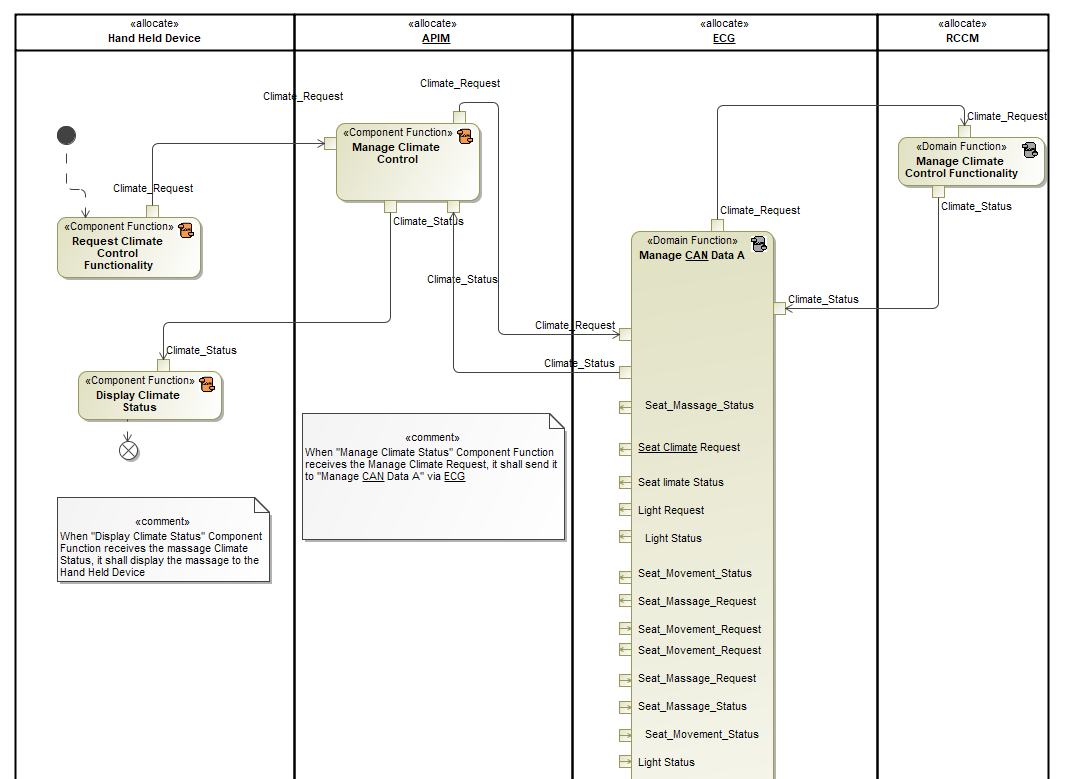


##### Function Interfaces

URC controls the quad zone climate system from a URC’s Users hand held device

URC User’s shall control only the climate zone they are seated in and when it is in quad zone mode.

**Climate Control Behavior:**



URC shall control climate using the existing climate CAN signals shown below:

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Quad Zone System** |  | **Sync to Climate** |  |  |  |  |
| **Seat Location** | **URC Logical Name** | **Signal Req (Climate Interface Spec)** | **Message** | **Signal** | **State** | **Comment** |
| LHS (Seat A) | Temp selection | REQ-389373 REQ-389384 | Clmt\_Button\_Stat4 | Frt\_Btn\_Status\_1st Frt\_Btn\_Status\_2nd | None\_Pressed |  |
| LHS\_Temp\_Inc\_Pressed |  |
| LHS\_Temp\_Dec\_Pressed |  |
| RHS (Seat B) | Temp selection | REQ-389376 REQ-389385 | Clmt\_Button\_Stat4 | Frt\_Btn\_Status\_1st Frt\_Btn\_Status\_2nd | None\_Pressed |  |
| RHS\_Temp\_Inc\_Pressed |  |
| RHS\_Temp\_Dec\_Pressed |  |
|  |  |  |  |  |  |  |
| Seat C | Rear Climate ON/OFF | REQ-389530 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Rear\_Power\_Pressed |  |
| Auto ON/OFF | REQ-389549 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| AUTO\_Pressed |  |
| Temp Selection | REQ-389502 REQ-389508 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| LHS\_Temp\_Inc\_Pressed |  |
| LHS\_Temp\_Dec\_Pressed |  |
| Fan Speed | REQ-389520 REQ-389525 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Blwr\_Inc\_Pressed |  |
| Blwr\_Dec\_Pressed |  |
| Vent location | REQ-389538 REQ-389542 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Floor\_Pressed |  |
|  |  |  |  |  |  |  |
| Seat D | Rear Climate ON/OFF | REQ-389532 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Third\_Rr\_Pwr\_Pressed | Third Rr = RHS |
| Auto ON/OFF | REQ-389551 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Third\_AUTO\_Pressed |  |
| Temp Selection | REQ-389503 REQ-389509 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| RHS\_Temp\_Inc\_Pressed |  |
| RHS\_Temp\_Dec\_Pressed |  |
| Fan Speed | REQ-389521 REQ-389526 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Third\_Blwr\_Inc\_Pressed |  |
| Third\_Blwr\_Dec\_Pressed |  |
| Vent location | REQ-389540 REQ-389544 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Third\_Floor\_Pressed |  |
|  |  |  |  |  |  |  |
| Seat E | Rear Climate ON/OFF | REQ-389530 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Rear\_Power\_Pressed |  |
| Auto ON/OFF | REQ-389549 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| AUTO\_Pressed |  |
| Temp Selection | REQ-389502 REQ-389508 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| LHS\_Temp\_Inc\_Pressed |  |
| LHS\_Temp\_Dec\_Pressed |  |
| Fan Speed | REQ-389520 REQ-389525 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Blwr\_Inc\_Pressed |  |
| Blwr\_Dec\_Pressed |  |
| Vent location | REQ-389538 REQ-389542 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Floor\_Pressed |  |
|  |  |  |  |  |  |  |
| Seat F | Rear Climate ON/OFF | REQ-389532 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Third\_Rr\_Pwr\_Pressed |  |
| Auto ON/OFF | REQ-389551 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Third\_AUTO\_Pressed |  |
| Temp Selection | REQ-389503 REQ-389509 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| RHS\_Temp\_Inc\_Pressed |  |
| RHS\_Temp\_Dec\_Pressed |  |
| Fan Speed | REQ-389521 REQ-389526 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Third\_Blwr\_Inc\_Pressed |  |
| Third\_Blwr\_Dec\_Pressed |  |
| Vent location | REQ-389540 REQ-389544 | Clmt\_Button\_Stat4 | Rr\_Btn\_Status\_1st Rr\_Btn\_Status\_2nd | None\_Pressed |  |
| Third\_Floor\_Pressed |  |

Table 5‑2: Input Signal mappings of Function Manage Seat Climate Request

###### Outputs

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Quad Zone System** |  | **Climate to Sync** |  |  |  |
| **Seat Location** | **URC Logical Name** | **Signal Req** | **Message** | **Signal** | **State** |
| LHS (Seat A) | Temp Selection | REQ-389366 | Clmt\_Button\_Stat2 | LHS\_Temp\_Display\_Digit1 | "ASCII" encoding |
| LHS\_Temp\_Display\_Digit2 | "ASCII" encoding |
| LHS\_Temp\_Display\_Digit3 | Off |
| \_0 |
| \_5 |
| Unused |
| EATC\_LHS\_Units | Off |
| Celsius |
| Farenheit |
| Unused |
| RHS (B) | Temp Selection | REQ-389367 | Clmt\_Button\_Stat2 | RHS\_Temp\_Display\_Digit1 | "ASCII" encoding |
| RHS\_Temp\_Display\_Digit2 | "ASCII" encoding |
| RHS\_Temp\_Display\_Digit3 | Off |
| \_0 |
| \_5 |
| Unused |
| EATC\_RHS\_Units | Off |
| Celsius |
| Farenheit |
| Unused |
|  |  |  |  |  |  |
| Seat C | Rear Climate On/Off | REQ-389531 | Clmt\_Button\_Stat3 | Rr\_Power\_Btn\_Stt | Enabled\_Inactive |
| Enabled\_Active |
| Disabled\_Inactive |
| Disabled\_Active |
| Auto On/Off | REQ-389550 | Clmt\_Button\_Stat3 | Rr\_AUTO\_Btn\_Stt | Enabled\_Inactive |
| Enabled\_Active |
| Disabled\_Inactive |
| Disabled\_Active |
| Temp Selection | REQ-389495 |  | Rr\_LHS\_Set\_Temp\_Dig1 Rr\_LHS\_Set\_Temp\_Dig2 |  |
| Fan Speed | REQ-389513 |  | Rr\_Blower\_Indicate\_Stat |  |
| Vent Location | REQ-389539 REQ-389543 |  | ClimtRearPnl\_B\_Stat |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| Seat D | Rear Climate On/Off | REQ-389533 | Clmt\_Button\_Stat7 | Rr\_Third\_Power\_Btn\_Stt | Enabled\_Inactive |
| Enabled\_Active |
| Disabled\_Inactive |
| Disabled\_Active |
| Auto On/Off | REQ-389552 | Clmt\_Button\_Stat7 | Rr\_Third\_AUTO\_Btn\_Stt | Enabled\_Inactive |
| Enabled\_Active |
| Disabled\_Inactive |
| Disabled\_Active |
| Temp Selection | REQ-389496 |  | Rr\_RHS\_Set\_Temp\_Dig1 Rr\_RHS\_Set\_Temp\_Dig2 |  |
| Fan Speed | REQ-389515 |  | Rr\_RHS\_Blower\_Indicate |  |
| Vent Location | REQ-389541 REQ-389545 |  | ClimtRearPnl3rd\_B\_Stat |  |
|  |  |  |  |  |  |
| Seat E | Rear Climate On/Off | REQ-389531 |  |  |  |
| Auto On/Off | REQ-389550 |  |  |  |
| Temp Selection | REQ-389495 |  |  |  |
| Fan Speed | REQ-389514 |  |  |  |
| Vent Location | REQ-389539 REQ-389543 |  |  |  |
|  |  |  |  |  |  |
| Seat F | Rear Climate On/Off | REQ-389533 |  |  |  |
| Auto On/Off | REQ-389552 |  |  |  |
| Temp Selection | REQ-389496 |  |  |  |
| Fan Speed | REQ-389515 |  |  |  |
| Vent Location | REQ-389541 REQ-389545 |  |  |  |

Table 5‑3: Output Signal mappings of Function Manage Seat Climate Request

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

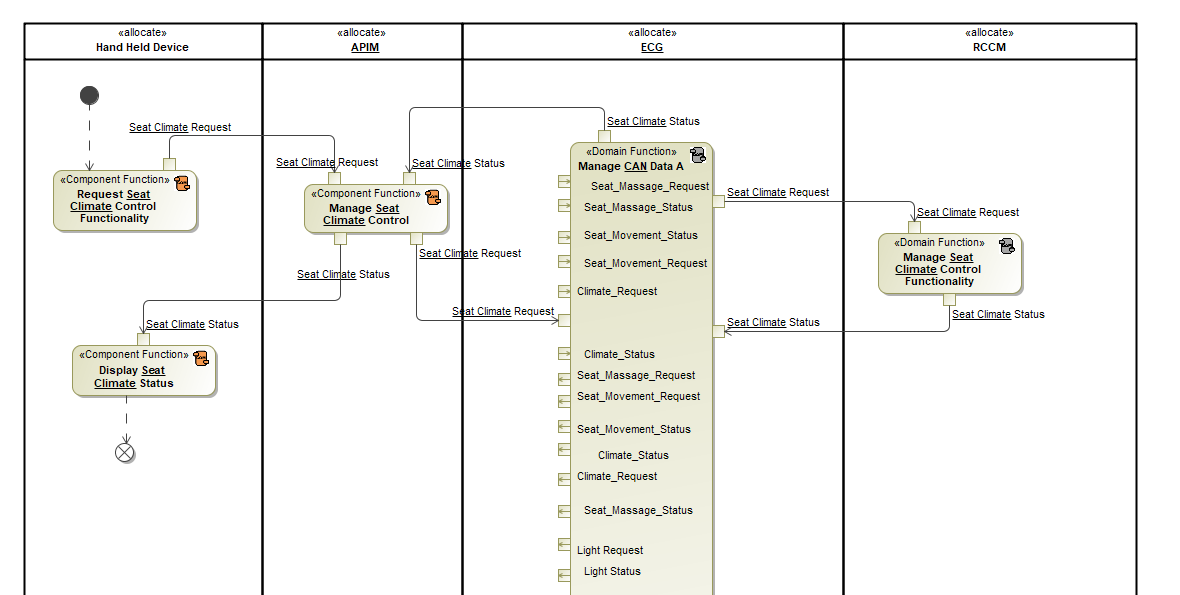
Table 5‑6: Inherited Requirements

###### Component Specific Requirements

#### Technology Function -340505036.jpg **Manage Seat Climate Control**

##### Function Interfaces

**Seat Climate Control Behavior:**



URC shall control climate using the existing seat climate CAN signals shown below:

###### Inputs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Seat Climate (Heat/Vent) |  |  |  |  |
| Driver (Seat A) | REQ-389477 | Clmt\_Button\_Stat4 | Frt\_Btn\_Status\_1st Frt\_Btn\_Status\_2nd | None\_Pressed |
| LHS\_Htd\_Seat\_Pressed |
| LHS\_Htd\_Seat1\_Pressed |
| LHS\_Htd\_Seat2\_Pressed |
| LHS\_Htd\_Seat3\_Pressed |
| LHS\_Cld\_Seat\_Pressed |
| LHS\_Cld\_Seat1\_Pressed |
| LHS\_Cld\_Seat2\_Pressed |
| LHS\_Cld\_Seat3\_Pressed |
| LHS\_Seat\_Off\_Pressed |
|  |  |  |  |  |
| Front Passenger (B) | REQ-389480 | Clmt\_Button\_Stat4 | Frt\_Btn\_Status\_1st Frt\_Btn\_Status\_2nd | None\_Pressed |
| RHS\_Htd\_Seat\_Pressed |
| RHS\_Htd\_Seat1\_Pressed |
| RHS\_Htd\_Seat2\_Pressed |
| RHS\_Htd\_Seat3\_Pressed |
| RHS\_Cld\_Seat\_Pressed |
| RHS\_Cld\_Seat1\_Pressed |
| RHS\_Cld\_Seat2\_Pressed |
| RHS\_Cld\_Seat3\_Pressed |
| RHS\_Seat\_Off\_Pressed |
|  |  |  |  |  |
| Seat C | REQ-401696 | Rear\_Climate\_Data2 | ClimtSeatHeatRl2\_B\_Stat | None\_Pressed |
| Pressed |
| ClimtSeatCoolRl2\_B\_Stat | None\_Pressed |
| Pressed |
|  |  |  |  |  |
| Seat D | REQ-401698 | Rear\_Climate\_Data2 | ClimtSeatHeatRr2\_B\_Stat | None\_Pressed |
| Pressed |
| ClimtSeatCoolRr2\_B\_Stat | None\_Pressed |
| Pressed |

Table 5‑2: Input Signal mappings of Function Manage Seat Climate Request

###### Outputs

Table 5‑3: Output Signal mappings of Function Manage Seat Climate Request

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

### OZM

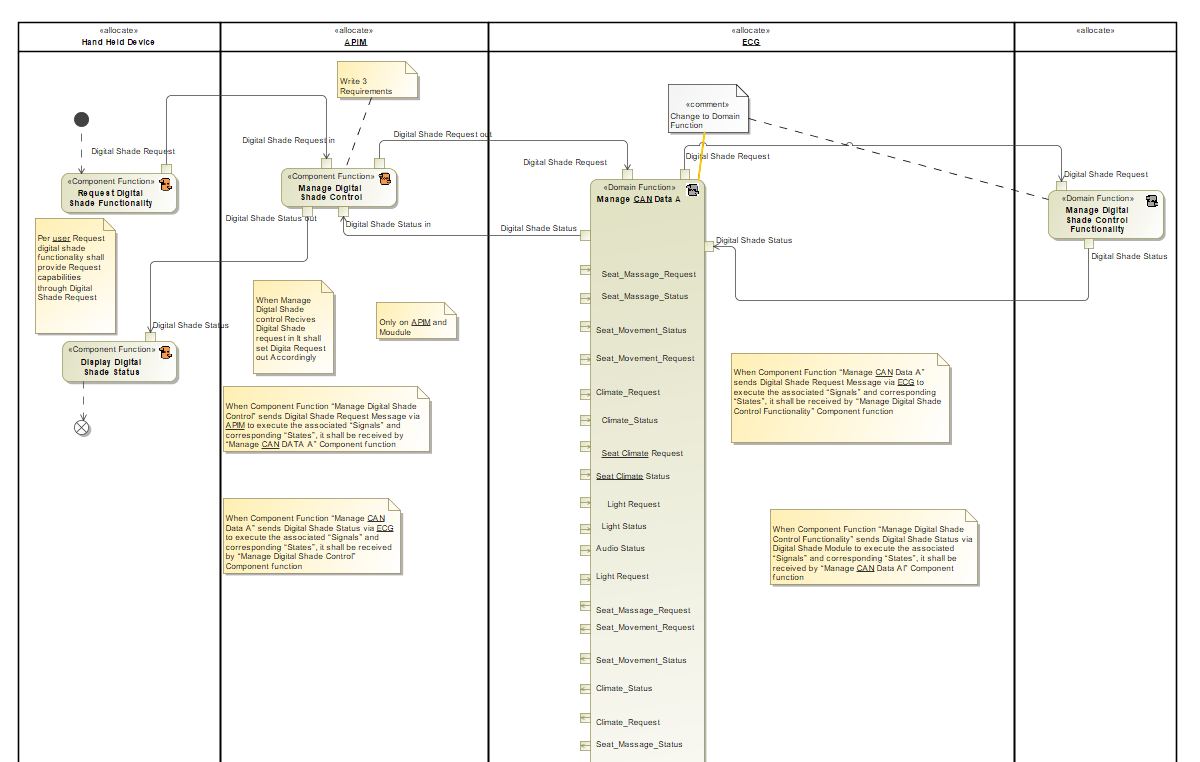
#### Technology Function 905376090.jpg **Mange Digital Shade Control**

##### Function Interfaces

See Digital Shade FIS document for more details

URC User’s shall control the area of digital shade above their seat only

**Digital Shade Control Behavior:**



###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| PSDS\_Mode\_Zone\_Rqst | 0x0 - Auto (default)  0x1 - Manual  0x2 - Zone\_1 (ON)  0x3 - Zone\_2  0x4 - Zone\_3  0x5 - Zone\_4  0x6 - Zone\_5  0x7 - Zone\_6  0x8 - Not\_Used | | |  | CAN |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| PSDS\_Mode\_Zone\_Status | TBD | | |  | CAN |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

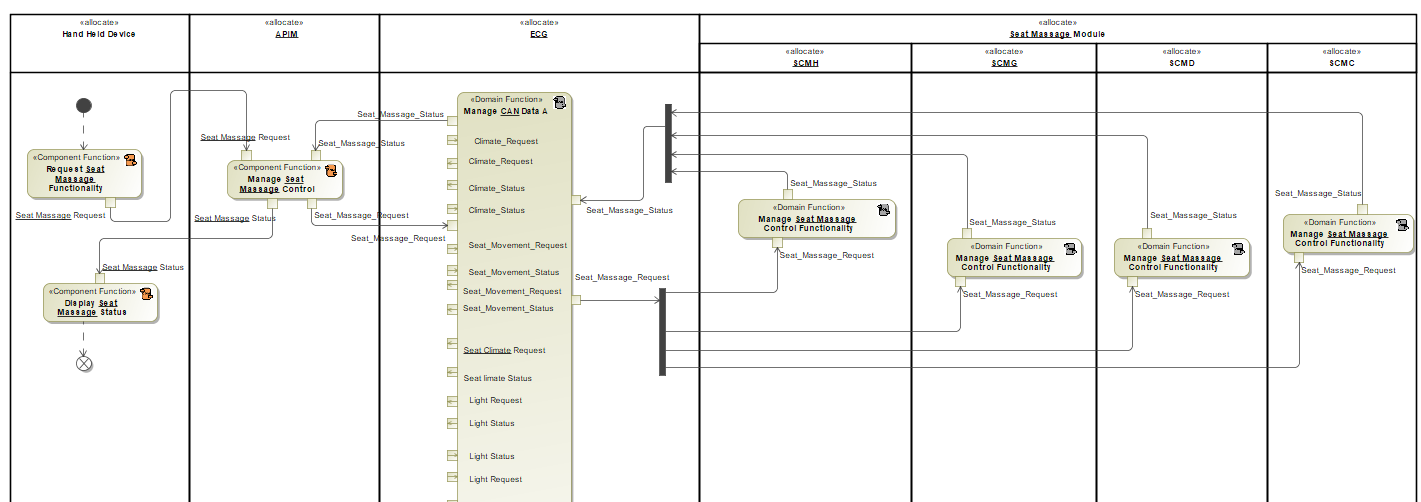
### SCMC

SCMC

#### Technology Function -1271660926.jpg **Manage 2R LH Seat Massage Request**

##### Function Interfaces

**Seat Massage Control Behavior (Using SoA Seat Service):**



1. URC shall use ECG Seat Services to communicate with SCMC for seat massage operation

2. SCMC shall consider URC Seat Request from ECG Service request as lowest priority

3. Reference SeatService VSEM ID: 735414

4. URC Shall utilize the Seat Service SoA interface to control seat movement with the APIs listed below

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Manage 2R LH Seat Massage Request | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_NONE, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_LOWER\_ROLLING, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_UPPER\_ROLLING, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_CUSHION, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_FULL\_RECOVERY, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_RELAXED\_RECOVERY, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_COMBO, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_ZIG\_ZAG\_COMBO, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_SWEEP\_WAVE, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_UP\_WAVE, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_PULSE, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_NONE, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_LOWER\_ROLLING, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_UPPER\_ROLLING, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_CUSHION, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_FULL\_RECOVERY, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_RELAXED\_RECOVERY, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_COMBO, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_ZIG\_ZAG\_COMBO, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_SWEEP\_WAVE, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_UP\_WAVE, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_PULSE, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_NONE, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_LOWER\_ROLLING, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_UPPER\_ROLLING, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_CUSHION, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_FULL\_RECOVERY, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_RELAXED\_RECOVERY, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_COMBO, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_ZIG\_ZAG\_COMBO, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_SWEEP\_WAVE, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_UP\_WAVE, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_DRIVER, MASSAGE\_ON, MASSAGE\_PULSE, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | Repeat APIs with MASSGE\_OFF | | |  | **Seat Service API** |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_DRIVER, NONE\_SELECTED, NULL\_ACTION) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_DRIVER, TOP\_LUMBAR, INCREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_DRIVER, TOP\_LUMBAR, DECREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_DRIVER, MIDDLE\_LUMBAR, INCREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_DRIVER, MIDDLE\_LUMBAR, DECREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_DRIVER, LOWER\_LUMBAR, INCREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_DRIVER, LOWER\_LUMBAR, DECREASE\_ONCE) | | |  | **Seat Service API** |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Seat Service Response Status | TBD | | |  | Placeholder |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

See SOA Interface Specification in VSEM for addition Seat Service requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

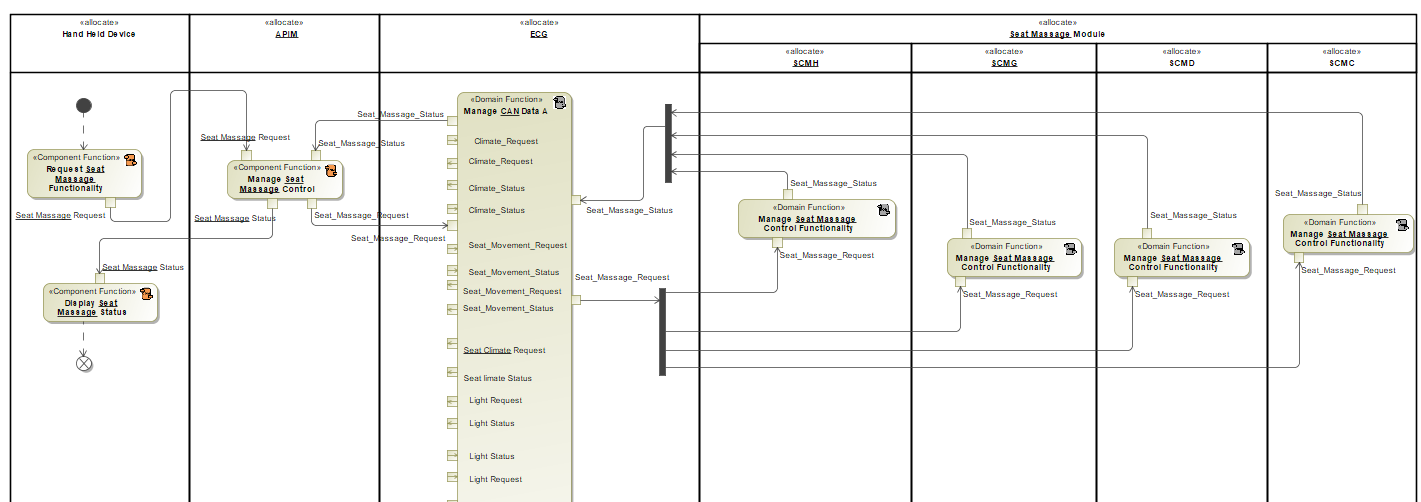
### SCMD

SCMD

#### Technology Function -254372083.jpg **Manage 2R RH Seat Massage Request**

##### Function Interfaces

**Seat Massage Control Behavior (Using SoA Seat Service):**



1. URC shall use ECG Seat Services to communicate with SCMD for seat massage operation

2. SCMD shall consider URC Seat Request from ECG Service request as lowest priority

3. Reference SeatService VSEM ID: 735414

4. URC Shall utilize the Seat Service SoA interface to control seat movement with the APIs listed below

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Manage 2R RH Seat Massage Request | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_NONE, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_LOWER\_ROLLING, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_UPPER\_ROLLING, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_CUSHION, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_FULL\_RECOVERY, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_RELAXED\_RECOVERY, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_COMBO, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_ZIG\_ZAG\_COMBO, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_SWEEP\_WAVE, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_UP\_WAVE, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_PULSE, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_NONE, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_LOWER\_ROLLING, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_UPPER\_ROLLING, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_CUSHION, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_FULL\_RECOVERY, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_RELAXED\_RECOVERY, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_COMBO, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_ZIG\_ZAG\_COMBO, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_SWEEP\_WAVE, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_UP\_WAVE, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_PULSE, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_NONE, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_LOWER\_ROLLING, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_UPPER\_ROLLING, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_CUSHION, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_FULL\_RECOVERY, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_RELAXED\_RECOVERY, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_COMBO, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_ZIG\_ZAG\_COMBO, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER MASSAGE\_ON, MASSAGE\_SWEEP\_WAVE, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_UP\_WAVE, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, SECOND\_ROW\_PASSENGER, MASSAGE\_ON, MASSAGE\_PULSE, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | Repeat APIs with MASSGE\_OFF | | |  | **Seat Service API** |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_PASSENGER, NONE\_SELECTED, NULL\_ACTION) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_PASSENGER, TOP\_LUMBAR, INCREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_PASSENGER, TOP\_LUMBAR, DECREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_PASSENGER, MIDDLE\_LUMBAR, INCREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_PASSENGER, MIDDLE\_LUMBAR, DECREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_PASSENGER, LOWER\_LUMBAR, INCREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, SECOND\_ROW\_PASSENGER, LOWER\_LUMBAR, DECREASE\_ONCE) | | |  | **Seat Service API** |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Seat Service Response Status | TBD | | |  | Placeholder |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

See SOA Interface Specification in VSEM for addition Seat Service requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
| REQ\_abc |  | Removed | -- |  |
| REQ\_def |  | Replaced | REQ\_xyz |  |
| -- |  | Added | REQ\_123 |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

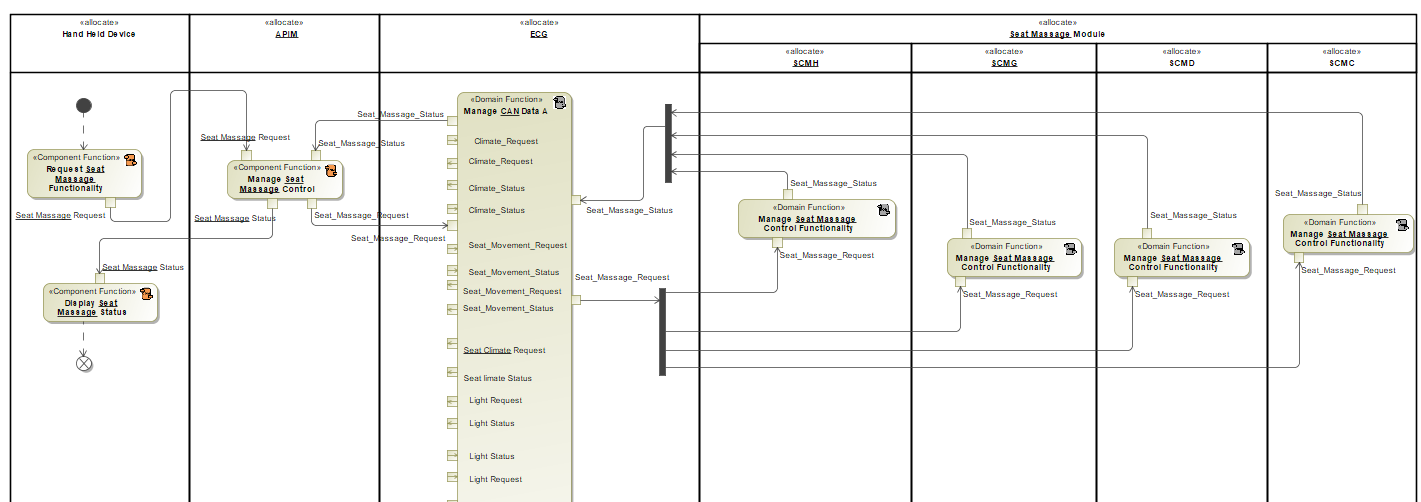
### SCMH

SCMH

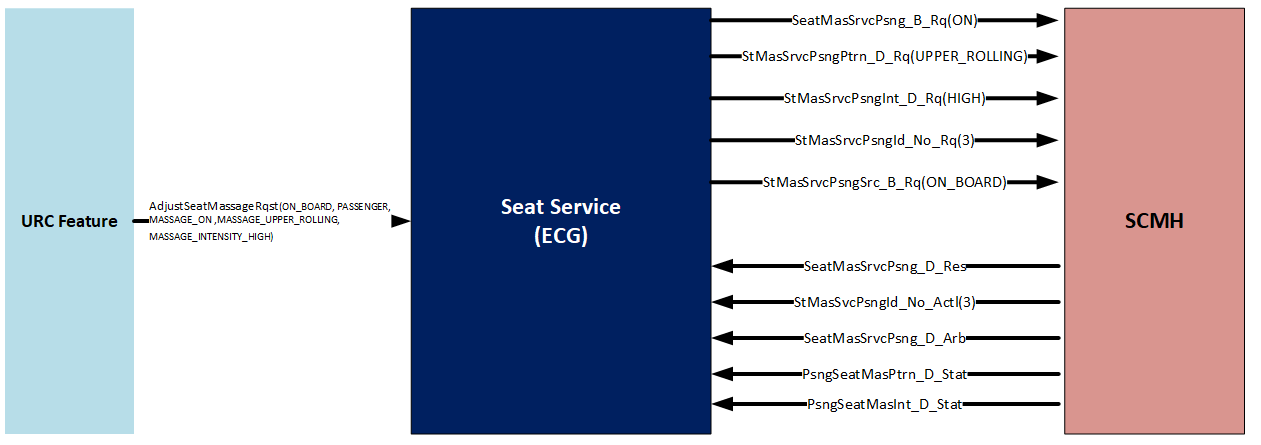
#### Technology Function 755055435.jpg **Manage 1R PASS Seat Massage Request**

##### Function Interfaces

**Seat Massage Control Behavior (Using SoA Seat Service):**



Example Flow:



1. URC shall use ECG Seat Services to communicate with SCMH for seat massage operation

2. SCMH shall consider URC Seat Request from ECG Service request as lowest priority

3. Reference SeatService VSEM ID: 735414

4. URC Shall utilize the Seat Service SoA interface to control seat movement with the APIs listed below

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Manage 2R LH Seat Massage Request | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_NONE, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_LOWER\_ROLLING, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_UPPER\_ROLLING, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_CUSHION, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_FULL\_RECOVERY, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_RELAXED\_RECOVERY, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_COMBO, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_ZIG\_ZAG\_COMBO, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_SWEEP\_WAVE, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_UP\_WAVE, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_PULSE, MASSAGE\_INTENSITY\_LOW) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_NONE, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_LOWER\_ROLLING, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_UPPER\_ROLLING, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_CUSHION, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_FULL\_RECOVERY, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_RELAXED\_RECOVERY, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_COMBO, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_ZIG\_ZAG\_COMBO, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_SWEEP\_WAVE, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_UP\_WAVE, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_PULSE, MASSAGE\_INTENSITY\_MEDIUM) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_NONE, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_LOWER\_ROLLING, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_UPPER\_ROLLING, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_CUSHION, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_FULL\_RECOVERY, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_RELAXED\_RECOVERY, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_CIRCULAR\_COMBO, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_ZIG\_ZAG\_COMBO, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_SWEEP\_WAVE, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_UP\_WAVE, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | AdjustSeatMassageRqst (OFF\_BOARD, PASSENGER, MASSAGE\_ON, MASSAGE\_PULSE, MASSAGE\_INTENSITY\_HIGH) | | |  | **Seat Service API** |  |
|  | Repeat APIs with MASSGE\_OFF | | |  | **Seat Service API** |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, PASSENGER, NONE\_SELECTED, NULL\_ACTION) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, PASSENGER, TOP\_LUMBAR, INCREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, PASSENGER, TOP\_LUMBAR, DECREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, PASSENGER, MIDDLE\_LUMBAR, INCREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, PASSENGER, MIDDLE\_LUMBAR, DECREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, PASSENGER, LOWER\_LUMBAR, INCREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, PASSENGER, LOWER\_LUMBAR, DECREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, PASSENGER, UPPER\_BOLSTER, INCREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, PASSENGER, UPPER\_BOLSTER, DECREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, PASSENGER, LOWER\_BOLSTER, INCREASE\_ONCE) | | |  | **Seat Service API** |  |
|  | AdjustBladderPressureRqst(OFF\_BOARD, PASSENGER, LOWER\_BOLSTER, DECREASE\_ONCE) | | |  | **Seat Service API** |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Seat Service Response Status | TBD | | |  | Placeholder |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

See SOA Interface Specification in VSEM for addition Seat Service requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

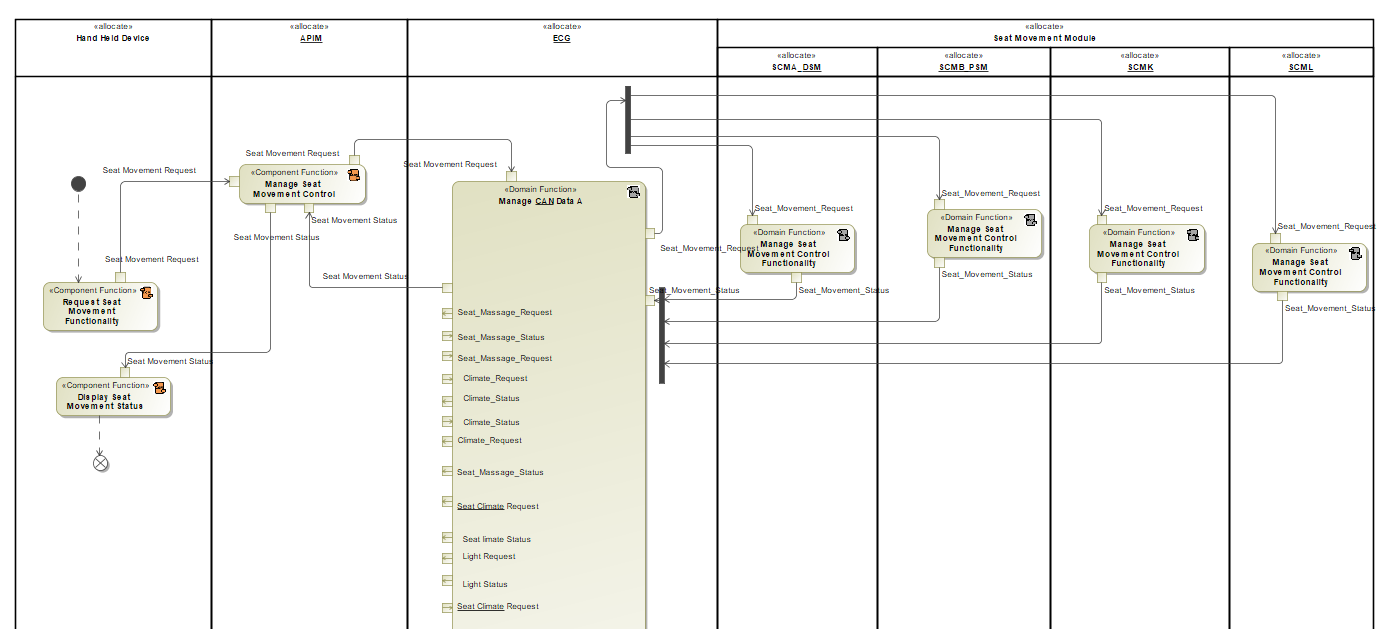
### SCMK

SCMK

#### Technology Function -1073815934.jpg **Manage Seat Movement 2R LH Request**

##### Function Interfaces

**Seat Position Control Behavior (Using SoA Seat Service):**



1. URC shall use ECG Seat Services to communicate with SCMK for seat control operation

2. SCMK shall consider URC Seat Request from ECG Service request as lowest priority

3. Reference SeatService VSEM ID: 735414

4. URC Shall utilize the Seat Service SoA interface to control seat movement with the APIs listed below

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Manage Seat Movement | MoveSeatRequest(OFF\_BOARD, SECOND\_ROW\_DRIVER, FORWARD\_UPWARD\_TAP, SEAT\_SLIDE) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, SECOND\_ROW\_DRIVER, REARWARD\_DOWNWARD\_TAP, SEAT\_SLIDE) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, SECOND\_ROW\_DRIVER, FORWARD\_UPWARD\_TAP, SEAT\_BACK\_INCLINATION) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, SECOND\_ROW\_DRIVER, REARWARD\_DOWNWARD\_TAP, SEAT\_BACK\_INCLINATION) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, SECOND\_ROW\_DRIVER, FORWARD\_UPWARD\_TAP, CALF\_REST) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, SECOND\_ROW\_DRIVER, REARWARD\_DOWNWARD\_TAP, CALF\_REST) | | |  | **Seat Service API** |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Seat Service Response Status | TBD | | |  | Placeholder |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

See SOA Interface Specification in VSEM for addition Seat Service requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
| REQ\_abc |  | Removed | -- |  |
| REQ\_def |  | Replaced | REQ\_xyz |  |
| -- |  | Added | REQ\_123 |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

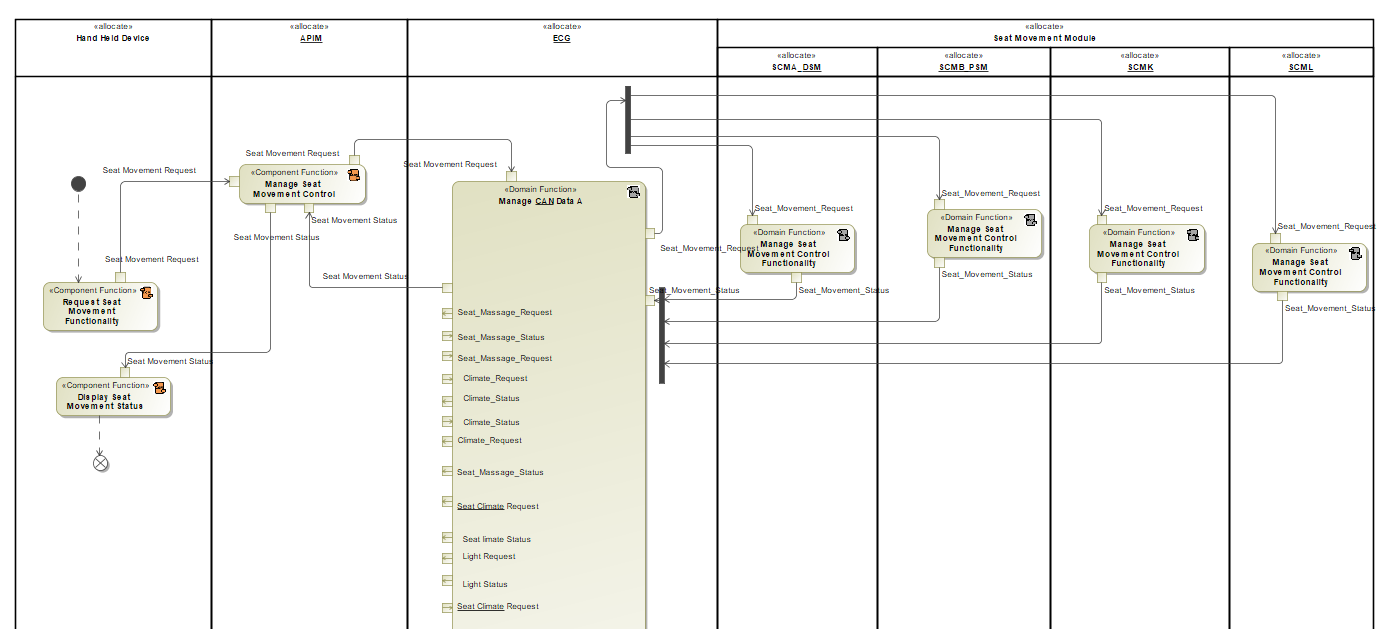
### SCML

SCML

#### Technology Function -1271660926.jpg **Manage Seat Movement 2R RH Request**

##### Function Interfaces

**Seat Position Control Behavior (Using SoA Seat Service):**



1. URC shall use ECG Seat Services to communicate with SCML for seat control operation

2. SCML shall consider URC Seat Request from ECG Service request as lowest priority

3. Reference SeatService VSEM ID: 735414

4. URC Shall utilize the Seat Service SoA interface to control seat movement with the APIs listed below

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Manage Seat Movement | MoveSeatRequest(OFF\_BOARD, SECOND\_ROW\_PASSENGER, FORWARD\_UPWARD\_TAP, SEAT\_SLIDE) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, SECOND\_ROW\_PASSENGER, REARWARD\_DOWNWARD\_TAP, SEAT\_SLIDE) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, SECOND\_ROW\_PASSENGER, FORWARD\_UPWARD\_TAP, SEAT\_BACK\_INCLINATION) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, SECOND\_ROW\_PASSENGER, REARWARD\_DOWNWARD\_TAP, SEAT\_BACK\_INCLINATION) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, SECOND\_ROW\_PASSENGER, FORWARD\_UPWARD\_TAP, CALF\_REST) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, SECOND\_ROW\_PASSENGER, REARWARD\_DOWNWARD\_TAP, CALF\_REST) | | |  | **Seat Service API** |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Seat Service Response Status | TBD | | |  | Placeholder |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

See SOA Interface Specification in VSEM for addition Seat Service requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
| REQ\_abc |  | Removed | -- |  |
| REQ\_def |  | Replaced | REQ\_xyz |  |
| -- |  | Added | REQ\_123 |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

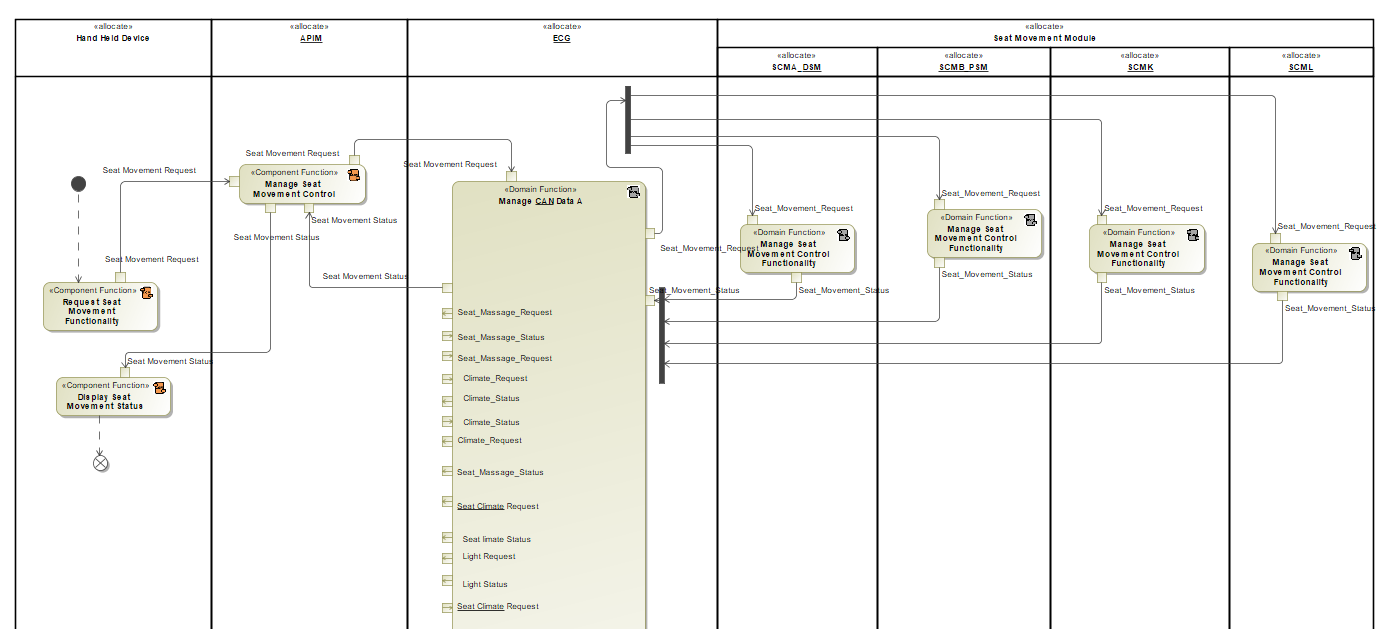
### SCMB\_PSM

SCMB\_PSM

#### Technology Function 344445004.jpg **Manage First Row Passenger Seat Movement Request**

##### Function Interfaces

**Seat Position Control Behavior (Using SoA Seat Service):**



1. URC shall use ECG Seat Services to communicate with SCMB\_PSM for seat control operation

2. SCMB\_PSM shall consider URC Seat Request from ECG Service request as lowest priority

3. Reference SeatService VSEM ID: 735414

4. URC Shall utilize the Seat Service SoA interface to control seat movement with the APIs listed below

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Manage Seat Movement | MoveSeatRequest(OFF\_BOARD, PASSENGER, FORWARD\_UPWARD\_TAP, SEAT\_SLIDE) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, REARWARD\_DOWNWARD\_TAP, SEAT\_SLIDE) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, FORWARD\_UPWARD\_TAP, SEAT\_REAR\_HEIGHT) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, REARWARD\_DOWNWARD\_TAP, SEAT\_REAR\_HEIGHT) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, FORWARD\_UPWARD\_TAP, SEAT\_FRONT\_HEIGHT) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, REARWARD\_DOWNWARD\_TAP, SEAT\_FRONT\_HEIGHT) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, FORWARD\_UPWARD\_TAP, SEAT\_BACK\_INCLINATION) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, REARWARD\_DOWNWARD\_TAP, SEAT\_BACK\_INCLINATION) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, FORWARD\_UPWARD\_TAP, RIGHT\_THIGH\_EXTEND) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, REARWARD\_DOWNWARD\_TAP, RIGHT\_THIGH\_EXTEND) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, FORWARD\_UPWARD\_TAP, LEFT\_THIGH\_EXTEND) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, REARWARD\_DOWNWARD\_TAP, LEFT\_THIGH\_EXTEND) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, FORWARD\_UPWARD\_TAP, SEAT\_BACKTOP\_INCLINATION) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, REARWARD\_DOWNWARD\_TAP, SEAT\_BACKTOP\_INCLINATION) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, FORWARD\_UPWARD\_TAP, HEADREST\_HEIGHT) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, REARWARD\_DOWNWARD\_TAP, HEADREST\_HEIGHT) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, FORWARD\_UPWARD\_TAP, HEADREST\_TILT) | | |  | **Seat Service API** |  |
|  | MoveSeatRequest(OFF\_BOARD, PASSENGER, REARWARD\_DOWNWARD\_TAP, HEADREST\_TILT) | | |  | **Seat Service API** |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Type** | **Connection**  (*Optional)* |
| Seat Service Response Status | TBD | | |  | Placeholder |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

See SOA Interface Specification in VSEM for addition Seat Service requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

### ECG

ECG Shall house the Seat Service that URC uses to request control of Seat Position and Seat Massage

#### Technology Function -1853935060.jpg **Manage CAN Data**

##### Function Interfaces

ECG will house the Seat Service for URC control of Seat Position and Seat Massage

ECG will be the gateway as normal for other CAN signals

###### Inputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Subscriber Interface** | **Connection**  (*Optional)* |
| TBD |  | | |  |  |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

Table 5‑2: Input Signal mappings of Function Manage CAN Data

###### Outputs

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | |
| **Logical Signal Name** | **Technical Signal Name** | | | **Mapping Details** *(Conditional)* | **Publisher Interface** | **Connection**  (*Optional)* |
| TBD |  | | |  |  |  |
|  |  | | |  |  |  |
|  |  | | |  |  |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | |

Table 5‑3: Output Signal mappings of Function Manage CAN Data

###### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Logical Parameter Name** | **Technical Parameter Name** | **Mapping Details** *(Conditional)* | **Method** | **Method Details** |
| Name should be a Word reference to the “*Logical Parameters*” name bookmark in the Data Dictionary | Name should be a Word reference to the “*Technical Parameters*” name bookmark in the Data Dictionary | If mapping is not 1:1 you might reference a Mapping description object from the *Mappings* section | Choose an item. | Depends on Method selection. For Method 2 a DID including start bit and length could be given. For Central Car Config a signal could be referenced |
|  |  |  |  |  |

Table 5‑4: Parameter mappings of Function “MyLogicalFunctionA\_Component1”

###### Interface Requirements

See SOA Interface Specification in VSEM for addition Seat Service requirements

##### Function Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Modification** | **Requirement ID**  (of Technology Function) | **Comment** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 5‑5: Component Specific Requirements

|  |  |  |
| --- | --- | --- |
| **Requirement ID**  (of Logical Function) | **Requirement Title** | **Comment** |
|  |  |  |
|  |  |  |
| … |  |  |

Table 5‑6: Inherited Requirements

###### Component Specific Requirements

## Requirements on Connections

### Networks

#### “CAN Bus xxx”

##### Protocol Requirements

##### Electrical Requirements

#### “LIN Bus xxx”

##### Protocol Requirements

###### Schedule Table

##### Electrical Requirements

#### “Ethernet xxx”

### HW I/Os

#### “HW I/O xxx”

## Requirements on Development Process

# Open Concerns

| ID | Concern Description | e-Tracker Reference | Status | Solution |
| --- | --- | --- | --- | --- |
| 1 | Mapping Bluetooth Low Energy signals to proper CAN/SoA signals |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |

Table 6‑1: Open Concerns

# Revision History

No Revision History found.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Revision | Date | Description | Approved by | Responsible |
| FIS REV 1 | 3/4/2021 | Initial version |  |  |
| FIS REV 2 | 6/8/2021 | Updated Signals and SoA APIs |  |  |

## Template Revisions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Rev. | Date | Description | Responsible |
| 0 | 2 | 2015-08-05 | * TOC corrected * Document Properties adapted to match needs of VBA macros | Awegman1 |
| 1 | 0 | 2015-11-16 | * Revision History moved to chapter 7 * Table-Styles removed | Awegman1 |
| 1 | 1 | 2016-03-02 | * Rework according to PCL example | Jbaden1 |
| 1 | 2 | 2016-03-22 | * V1.3: Footer formating corrected (Issue 19) * “Constraints” chapter renamed to “Input Requirements” (Issue 20) | Jbaden1 |
| 1 | 3 | 2016-04-20 | * Broken Wiki links repaired | Jbaden1 |
| 2 | 0 | 2016-05-23 | * Prepared for Specification\_Macros.dotm v2.0 * Additional explanations added to ch. 2.2 “Input Requirements” (ARL and SDS requirements often go here) | Jbaden1 |
| 2 | 1 | 2016-07-08 | * Template version added to footer | Jbaden1 |
| 2 | 2 | 2016-07-15 | * Sample SysML diagrams added * Data Dictionary reworked * Alignment with relevant sections in SRD templated | Jbaden1 |
| 3 | 0 | 2016-09-05 | * Lessons learned from IPRB incorporated | Jbaden1 |
| 4 | 0 | 2016-09-27 | * Alignment with QPIP Feature Function Ownership workstream. Platform Spec renamed to Feature Implementation Spec | Jbaden1 |
| 4 | 1 | 2016-11-04 | * Chapters “Purpose” and “Scope” reworked. | Jbaden1 |
| 4 | 1 | 2016-11-10 | * Subsection for “Logical Service Interfaces” added. | Jbaden1 |
| 5 | 0 | 2017-01-13 | * Meta data updated for specification macros, version 3.1 * SW Unit chapter removed for the time being * Green boxes added for user hints | Jbaden1 |
| 5 | 1 | 2017-01-18 | * Minor editorial changes (e.g. hyperlinks highlighted in comments) | Jbaden1 |
| 5 | 1b | 2017-01-20 | * Some editorial corrections * Substructure of old Network Communication (now Connections) moved to Requirements on Connections | Jbaden1 |
| 6 | 0 | 2018-07-24 | * CR53: * Add new cover sheet * Add disclaimer section * Add the following meta-data to the doc properties for the the new cover sheet   + DocGis1ItemNumber   + DocGis2Classification   + DocType   + DocStatus   + DocIssueDate   + DocReleaseDate * CR63: Update FuSa sharepoint references in templates | Jbaden1 |
| 6 | 0 | 2018-08-06 | * CR81: Incorporate lessons learned from System Service Spec pilot (Vehicle Speed) into AFS and FIS | Jbaden1 |
| 6 | 0 | 2018-09-28 | * Broken links to RE Wiki repaired | Jbaden1 |
| 6 | 0 | 2018-10-31 | * Minor corrections on cover sheet and in footer to be more GIS compliant and VSEM aligned * “Overview” and “Description” exchanged in headings (following common sense) | Jbaden1 |
| 6 | 0 | 2018-11-30 | * Update of Functional Safety sections after review by Functional Safety Team * Initial support for variant handling | Jbaden1 |
| 6 | 0 | 2018-12-01 | * Variant condition fields added consistently * Links updated | Jbaden1 |
| 6 | 0 | 2018-12-11 | * Variant condition fields removed from mapping/allocation tables * Mapping tables simplified * Explanatory text for “Variants” sections revised | Jbaden1 |
| 6 | 0a | 2019-01-04 | * Chapter heading “Inherited Function Requirements” removed. Corresponding table renamed to “Requirements not cascaded”. * E/E Connection table got another column for allocated messages * Naming conventions for Implemented Functions corrected (FncName\_CmpName instead of FncName\_on\_CmpName) * Editorial corrections on the cover sheet * Explanatory text added to “Ethernet” section in chapter “Requirements on Connections” * AIS templates updated. Linked to Wiki page | Jbaden1 |
| 6 | 0a | 2019-01-04 | * Minor restructuring in FuSa chapter – after aligning with ECU Functional Spec * Bugfix: table 13 renamed from FTTI table to FHT table, includes a bug fix: each FSR is allocated to only one ECU/component | Jbaden1 |
| 6 | 0b | 2019-02-04 | * Change: Chapter “Interface Requirements” added to “Implemented Function xxx” section (to have a single chapter for to collect subscriber/publisher interface and mapping requirements which to not conform to the corresponding Data Dictionary objects) * Change: “CAN Interface” subsection renamed to “AIS Interfaces” again. Although several Subscriber/Publisher interface attributes are probably CAN bus specific, other attributes seem to be well suited for other networks than CAN. * Change: Chapter “ECU Specific Requirements” renamed to “Component Specific Requirements” in chapter “Implemented Function xxx”. Table “Requirements not cascaded” renamed to “Component Specific Requirements” and refined to describe changes from Logical Function requirements set more formally. This is also to help during VSEM import to identify those requirements of the Logical Function which cannot be simply carried over to the ECU. * Change: Explanatory text in section “Implemented Function xxx” improved. | Jbaden1 |
| 6 | 0c | 2019-02-05 | * Change: Layout of AIS Interfaces in Data Dictionary reworked to enable Excel Import | Jbaden1 |
| 6 | 0c | 2019-02-20 | * Bugfix: In AIS Interfaces none-picklist fields formatted as invisible | Jbaden1 |
| 6 | 1a | 2019-02-05 | Functional Safety related changes:   * Table “Architectural Redundancy Summary” updated * Section “Functional Flows for FTTI ‘xyz’” added to chapter “Component Interaction Diagrams” * Fault Tolerant Time Summary section added to Functional Safety chapter * Chapter “HW Metrics” added | Jbaden1 |
| 6 | 1a | 2019-04-02 | Headings of “Architectural Redundancy Summary” table clarified | Jbaden1 |
| 6 | 1a | 2019-04-10 | * ASIL Decomposition table moved from Function Spec into the Feature Implementation Spec (ASIL Decomposition of Technical Safety Requirements) * 2 alternative versions of the Function Allocation Table (Standard variant vs. Functional Safety variant) placed next to each other. | Jbaden1 |
| 6 | 1a | 2019-05-31 | * Function Allocation Table split into a base (non FuSa) part and a FuSa part to allow a more flexible mapping of MBSE functions (Logical and Technology) to RE functions (Atomic Logical and Implemented). | Jbaden1 |
| 6 | 1a | 2019-05-31 | * “Input Requirement” section reworked (symmetrically to all other templates). * Sections “Functional Flows for FTTI xyz” and “Fault Tolerant Time Summary” removed, because guidance is not available yet. * “Reference” and “Glossary” section moved back to introduction, i.e., to the very beginning of the document (such that also section 2 can already rely on it). * Some mostly editorial changes per request from FuSa team. | Jbaden1 |
| 6 | 1a | 2019-07-02 | * "Important" box added on cover sheet which points to the macros * “Input Requirements” section renamed to Input Information (after discussion with FuSa team) | Jbaden1 |
| 6 | 1a | 2019-07-17 | * Chapter “Message List” removed from CAN and LIN specific chapters of section “Requirements on Connections” | Jbaden1 |
| 6 | 1a | 2019-10-08 | * Chapter “ASIL Decomposition of Technical Safety Requirements”: Input TSRs are specified in the chapter right above the decomposition table. | Jbaden1 |
| 6 | 1a | 2019-10-09 | * Chapter “Service Oriented Communication” moved to section “Messages” in the Data Dictionary. Details from Central SW Wiki about FNV2 SOA added | Jbaden1 |
| 6 | 1a | 2019-10-25 | * Minor updates for HW IOs/Signals * Subsection “Functional Safety” removed from chapter “Feature Implementation Modeling”. Per requrest from FuSa team since no guidance is available how to model e.g. FHT timing diagram. | Jbaden1 |
| 6 | 1a | 2019-05-11 | * Copyright notice shortened and moved to cover sheet and added to footer (to be compliant [with Ford copyright guidelines](http://www.fgti.ford.com/client/NewFGTI/CopyrightNotice.html)) * Term “Disclaimer” no longer used for what is actually only a copyright notice | Jbaden1 |
| 6 | 1a | 2019-22-11 | * Some minor modifications for the SOA APIs/MQTT Messages in the section “Messages” of the Data Dictionary (section references Service Contracts via the API name) * Some minor updates of the Input/Output mapping tables in section “Requirements on Components” for mappings to SOA APIs and EDAS signals. | Jbaden1 |
| 6 | 1a | 2019-12-05 | * Upstream Documents section added to “Input Requirements/Documents” table * Custom style table formatting removed | Jbaden1 |
| 6 | 1a | 2020-01-07 | * Some fine tuning for naming conventions of E/E components and connections. * List of HW I/O signal types reduced to RF-A, RF-D, D, A, Networked and PWM. * Protocol column added to the E/E connection table | Jbaden1 |
| 6 | 1a | 2020-01-07 | * “HW Metric” and “Architecture Redundancy Summary” sections removed per request from the Functional Architecture Team (based on Governance Board decision [FSTGB-97](mailto:TrackLite%20%23%20FSTGB-97:%20https://www.tracklite.ford.com/prweb/PRAuth/TrackLiteSSO?pyActivity=@baseclass.RedirectAndRunWraper&ThreadName=WorkLinkThread&bPurgeTargetThread=true&AccessGroupName=FSTGB:ProjectAdministrators&Location=pyActivity%3DWork-.Open%26Action%3DReview%26HarnessPurpose%3DReview%26InsHandle%3DFORD-FSTGB-WORK+FSTGB-97)) * “Functional Safety” chapter moved to “Feature Implementation Requirements” section. “Function Allocation” chapter seemed no longer appropriate. | Jbaden1 |
| 6 | 1a | 2020-01-07 | * Ordering of fields in AIS interfaces tables modified to conform with the Macro Template and the Importer Sheet * Page Header: no longer in bold letters | Jbaden1 |
| 6 | 1a | 2020-03-09 | * Missing doc property “LatestSigMappingID” and “LatestAisInterfaceID” added * doc property “CopyrightDate” re-formatted to text and copyright date field in footer corrected * Version numbering re-initialized as 0.1 * Init value of version/revision date set to “yyyy/mm/dd” instead of “yyyy-mm-dd” to be in line with the “Edit Document Property” dialog * Type of “Latest….ID” doc properties changed from Text to Number | Jbaden1 |
| 6 | 1a | 2020-03-11 | * “Mapping” table removed from template. Has been migrated to macro. | Jbaden1 |
| 6 | 1a | 2020-03-13 | * Separate chapter “Technical Safety Requirements” removed. Content already covered by Allocation Table in chapter Function Allocation. * “Implemented Function” replaced by term “Technology Function” | Jbaden1 |

# Appendix

## Data Dictionary

### Logical Signals

See URC Function Specification for detailed Logical Signals.

User Feedback

|  |  |  |
| --- | --- | --- |
| **ASIL** | | Choose an item. |
| **Encoding Type Name** | |  |
| Note: An encoding is either discrete or continuous. Delete fields below which are not needed | | |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) | Value 1 |  |
| Value 2 | … |
| … | … |
|  |  |
|  |  |
| **Unit** | |  |

Table: Signal Details of User Feedback

### Logical Parameters

### Technical Signals

Level 1\_2\_3\_4

|  |  |  |
| --- | --- | --- |
| **ASIL** | | Choose an item. |
| **Init Default Value** | |  |
| **Encoding Type Name** | |  |
| Note: An encoding is either discrete or continuous. Delete fields below which are not needed, | | |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) | Value 1 |  |
| Value 2 | … |
| … | … |
|  |  |
|  |  |
| **Unit** | |  |

Table: Signal Details of Level 1\_2\_3\_4

#### GSDB Signals

#### HW I/Os

#### Diagnostic Interfaces

##### DTCs

<Some Description of the DTC.

Refer to VSEM document “[Diagnostic Fault Coverage and DTC Numbers](https://www.vsemweb.ford.com/tc/launchapp?-attach=true&-s=226TCSession&-o=yAUtrNhnx3NrTDAAAAAAAAAAAAA&servername=Production_Server)

[Design Consideration](https://www.vsemweb.ford.com/tc/launchapp?-attach=true&-s=226TCSession&-o=yAUtrNhnx3NrTDAAAAAAAAAAAAA&servername=Production_Server)”, what to fill into the attributes below>

|  |  |
| --- | --- |
| **Test Period Time** |  |
| **Test Run Criteria,** |  |
| **Enable Criteria (EC)** |  |
| **Applicable** |  |
| **FailureTypeBytes** |  |
| **Test Period Time** |  |
| **Test Run Criteria,** |  |

##### DIDs

### Technical Parameters

### Mappings

### Technical Interfaces

#### AIS Interfaces

##### Publisher Interfaces

##### Subscriber Interfaces

#### AUTOSAR Ports

### Messages/APIs

#### CAN Bus “<Bus Name>”

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CAN ID** | **Transmission Mode** | **Period** | **Signal Names** | **Transmitter(s)** | **Receiver(s)** |
|  |  |  |  |  |  |
|  |
|  |
|  |

#### LIN Bus “<Bus Name>”

#### AUTOSAR Interfaces

#### SOA Service Contracts

<Service contract purpose/behavior>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Messaging Pattern | Frequency  (For Data Broadcast Only) | Message Data Element(s)  (Must Match GPB) or applicable CAN signal | Description of Data Element(s) | Topic Name |
| Choose an item. |  | GBP Data element / CAN Signal name 1 | Detailed encoding of data element 1 |  |
| … |  |  |
| GBP Data element / CAN Signal name 1 | Detailed encoding of data element 3 |  |

### Encoding Types

Seat\_Climate\_Request

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **AdjustSeatClimateRqst\_URC\_DRIVER\_climate\_mode** |  |
| **AdjustSeatClimateRqst\_URC\_DRIVER\_et\_points** |  |
| **AdjustSeatClimateRqst\_URC\_1R PASS\_climate\_mode** |  |
| **AdjustSeatClimateRqst\_URC\_1R PASS\_set\_points** |  |
| **AdjustSeatClimateRqst\_URC\_2R LH\_climate\_mode** |  |
| **AdjustSeatClimateRqst\_URC\_2R LH\_set\_points** |  |
| **AdjustSeatClimateRqst\_URC\_2\_ RH\_climate\_mode** |  |
| **AdjustSeatClimateRqst\_URC\_2R RH\_set\_points** |  |
| **Unit** | |  |

Table: Encoding Details of Seat\_Climate\_Request

AudioSource

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **SOURCE\_1** |  |
| **SOURCE\_2** |  |
| **SOURCE\_3** |  |
| **Unit** | |  |

Table: Encoding Details of AudioSource

DriverAcceptance

"DriverAcceptance" signal is for the driver to ACCEPT or REJECT the user to use the URC

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **ACCEPT** |  |
| **REJECT** |  |
| **Unit** | |  |

Table: Encoding Details of DriverAcceptance

URCConnectionStatus

"URCConnectionStatus" signal identify if URC is connected to vehicle or not via "URC\_CONNECTED\_VEHICLE" and "URC\_NOT\_CONNECTED\_VEHICLE"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **URC\_CONNECTED\_VEHICLE** |  |
| **URC\_NOT\_CONNECTED\_VEHICLE** |  |
| **Unit** | |  |

Table: Encoding Details of URCConnectionStatus

DisconnectURC

“DisconnectURC” signal to identify if URC user is disconnected or not via YES or NO

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **YES** |  |
| **NO** |  |
| **Unit** | |  |

Table: Encoding Details of DisconnectURC

ZoneClimateSettingsStatus

"ClimateSettingsStatus" display Climate settings status via "UPDATE\_SUCCESSFUL","VENTING\_DISTRIBUITION\_FAILED","DESIRED\_TEMPERTATURE\_FAILED","NOT\_APPLICABLE","TEMPERATURE\_UNIT\_FAILED","FAN\_SPEED\_FAILED"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **UPDATE\_SUCCESSFUL** |  |
| **VENTING\_DISTRIBUITION\_FAILED** |  |
| **DESIRED\_TEMPERTATURE\_FAILED** |  |
| **NOT\_APPLICABLE** |  |
| **TEMPERATURE\_UNIT\_FAILED** |  |
| **FAN\_SPEED\_FAILED** |  |
| **Unit** | |  |

Table: Encoding Details of ZoneClimateSettingsStatus

Light\_Request

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Request\_Light\_On/Off** |  |
| **Request\_LightAmbColor\_No\_Rq** |  |
| **Request\_LightAmbIntsty\_No\_Rq** |  |
| **Unit** | |  |

Table: Encoding Details of Light\_Request

Climate\_Status

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Request\_Temp\_Quad1\_Status** |  |
| **Request\_Temp\_Quad2\_Status** |  |
| **Request\_Temp\_Quad3\_Status** |  |
| **Request\_Power\_On/Off\_Quad3\_Status** |  |
| **Request\_Auto\_On/Off\_Quad3\_Status** |  |
| **Request\_Fan\_Speed\_Quad3\_Status** |  |
| **Request\_Vent\_Mode\_Quad3\_Status** |  |
| **Request\_Power\_On/Off\_Quad4\_Status** |  |
| **Request\_Auto\_On/Off\_Quad4\_Status** |  |
| **Request\_Fan\_Speed\_Quad4\_Status** |  |
| **Request\_Vent\_Mode\_Quad4\_Status** |  |
| **Request\_Vent\_Mode\_Quad3\_Status** |  |
| **Request\_Power\_On/Off\_Quad4\_Status** |  |
| **Request\_Auto\_On/Off\_Quad4\_Status** |  |
| **Request\_Fan\_Speed\_Quad4\_Status** |  |
| **Unit** | |  |

Table: Encoding Details of Climate\_Status

UserRequestSeatZoneAction

“UserRequestSeatZoneAction” signal define URC user request seat zone

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of UserRequestSeatZoneAction

RejuvenateInhibitFunctionality

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **INHIBIT\_DRIVER\_URC\_AUDIO** |  |
| **INHIBIT\_DRIVER\_URC\_INTERIOR\_LIGHTING** |  |
| **INHIBIT\_DRIVER\_URC\_DIGITAL\_SHADE** |  |
| **Unit** | |  |

Table: Encoding Details of RejuvenateInhibitFunctionality

DriverAcceptanceAction

“DriverAcceptanceAction” signal for the driver acceptance action to accept user URC connection to vehicle or not

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of DriverAcceptanceAction

Pattern

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **PATTERN1** |  |
| **PATTERN2** |  |
| **PATTERN3** |  |
| **Unit** | |  |

Table: Encoding Details of Pattern

DriverVerifyUserPinAction

“DriverVerifyUserPinAction” signal for the driver to verify URC user pin

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of DriverVerifyUserPinAction

Seat\_Climate\_Status

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **AdjustSeatClimateRqst\_URC\_DRIVER\_climate\_mode\_Status** |  |
| **AdjustSeatClimateRqst\_URC\_DRIVER\_et\_points\_Status** |  |
| **AdjustSeatClimateRqst\_URC\_1R PASS\_climate\_mode\_Status** |  |
| **AdjustSeatClimateRqst\_URC\_1R PASS\_set\_points\_Status** |  |
| **AdjustSeatClimateRqst\_URC\_2R LH\_climate\_mode\_Status** |  |
| **AdjustSeatClimateRqst\_URC\_2R LH\_set\_points\_Status** |  |
| **AdjustSeatClimateRqst\_URC\_2\_ RH\_climate\_mode\_Status** |  |
| **AdjustSeatClimateRqst\_URC\_2R RH\_set\_points\_Status** |  |
| **Unit** | |  |

Table: Encoding Details of Seat\_Climate\_Status

DriverApprovalRequeset

"DriverApprovalRequeset" signal to request driver approval via "REQUESTAPPROVAL" or "NONE"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **REQUESTAPPROVAL** |  |
| **NONE** |  |
| **Unit** | |  |

Table: Encoding Details of DriverApprovalRequeset

LEFT\_SECON\_ROW

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **INHIBIT\_CLIMATE** |  |
| **INHIBIT\_GLOBAL\_AUDIO** |  |
| **INHIBIT\_INERIOR\_LIGHTING** |  |
| **INHIBIT\_DIGITAL\_SHADE** |  |
| **Unit** | |  |

Table: Encoding Details of LEFT\_SECON\_ROW

SeatVenting

“SeatVenting” signal define seat venting settings via Zero\_PRCNT, TEN\_PRCNT, ……., HUNDRED\_PRCNT

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Zero\_PRCNT** |  |
| **TEN\_PRCNT** |  |
| **TWENTY\_PRCNT** |  |
| **THIRTY\_PRCNT** |  |
| **FOURTY\_PRCNT** |  |
| **FIFTY\_PRCNT** |  |
| **SIXTY\_PRCNT** |  |
| **SEVENTY\_PRCNT** |  |
| **EIGHTY\_PRCNT** |  |
| **NINTY\_PRCNT** |  |
| **HUNDRED\_PRCNT** |  |
| **Unit** | |  |

Table: Encoding Details of SeatVenting

LightCommand

"LightCommand" signal determine "LightingScheme", "LightColor","LightIntensity","LightONOFF"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of LightCommand

DIGITAL\_SHADE\_LOCKOUT\_STATUS

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **LOCKED** |  |
| **NOT\_LOCKED** |  |
| **Unit** | |  |

Table: Encoding Details of DIGITAL\_SHADE\_LOCKOUT\_STATUS

DesiredTemperature

"DesiredTemperature" signal select the desired zone temperature

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of DesiredTemperature

SEAT\_MOVEMENT

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **AVAILABLE** |  |
| **UNAVAILABLE** |  |
| **Unit** | |  |

Table: Encoding Details of SEAT\_MOVEMENT

LightingScheme

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **SCHEME1** |  |
| **SCHEME2** |  |
| **Unit** | |  |

Table: Encoding Details of LightingScheme

SeatPositionSettingsStatus

"SeatPositionSettingsStatus" signal provide the seat position settings status via "UPDATE\_SUCCESSFUL","SEAT\_CUSHION\_MOVEMENT\_UPDATE\_FAILED","SEAT\_BACK\_MOVEMENT\_UPDATE\_FAILED","SEAT\_REST\_MOVEMENT\_UPDATE\_FAILED","BLADDERS\_UPDATE\_FAILED","NOT\_APPLICABLE"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **UPDATE\_SUCCESSFUL** |  |
| **SEAT\_CUSHION\_MOVEMENT\_UPDATE\_FAILED** |  |
| **SEAT\_BACK\_MOVEMENT\_UPDATE\_FAILED** |  |
| **SEAT\_REST\_MOVEMENT\_UPDATE\_FAILED** |  |
| **BLADDERS\_UPDATE\_FAILED** |  |
| **NOT\_APPLICABLE** |  |
| **Unit** | |  |

Table: Encoding Details of SeatPositionSettingsStatus

AudioAction

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **PAUSE** |  |
| **SEEK** |  |
| **PLAY** |  |
| **Unit** | |  |

Table: Encoding Details of AudioAction

SEAT\_MASSAGE

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **AVAILABLE** |  |
| **UNAVAILABLE** |  |
| **Unit** | |  |

Table: Encoding Details of SEAT\_MASSAGE

SEAT\_MOVEMENT\_LOCKOUT\_STATUS

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **LOCKED** |  |
| **NOT\_LOCKED** |  |
| **Unit** | |  |

Table: Encoding Details of SEAT\_MOVEMENT\_LOCKOUT\_STATUS

HMI\_Feedback

"HMI\_Feedback" signal display HMI feedback via Sub-Signals;"URCDisplayStatus ","URCConnectionStatus”, "InhibitSettings“, “UserLeaveVehicle”, “ConnectedSeaID”

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of HMI\_Feedback

SeatBackMovement

“SeatBackMovement ” signal define seat back movement settings via BACK\_RECLINE, BACK\_UPPER\_PIVOT, LUMBAR\_MECHANICAL, LUMBAR\_BLADDERS

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **BACK\_RECLINE** |  |
| **BACK\_UPPER\_PIVOT** |  |
| **LUMBAR\_MECHANICAL** |  |
| **LUMBAR\_BLADDERS** |  |
| **Unit** | |  |

Table: Encoding Details of SeatBackMovement

UserHMIPinConnectionAction

“UserHMIPinConnectionAction” signal for the URC user to accept provided connection pin

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of UserHMIPinConnectionAction

Audio\_Request

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Request\_Volume\_Change** |  |
| **Request\_Source\_Change** |  |
| **Request\_Seek\_Change** |  |
| **Request\_Audio\_Share** |  |
| **Request\_ICC** |  |
| **Request\_DnD** |  |
| **Request\_Zone\_Volume\_Change** |  |
| **Unit** | |  |

Table: Encoding Details of Audio\_Request

InhibitSettings

"InhibitSettings" signal inhibit URC functionality via "InhibitFunctionality" that include "INHIBIT\_LIGHTING","INHIBIT\_CLIMATE"," INHIBIT\_AUDIO","INHIBIT\_SEATING\_POSITION","NONE"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of InhibitSettings

SeatAudioSettings

SeatAudioSettings" signal identify; seat Audio settings via Sub-signal "SeatAudioLevelPercent

" that specify audio volume levels through "0\_PERCENT, 10\_ PERCENT

, ….100\_PERCENT" and sub-signal SeatAudioActions

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of SeatAudioSettings

LockoutFunctionality

“LockoutFunctionality” signal contains 8 sub signal (1) INTERIOR\_LIGHTING\_LOCKOUT\_STATUS ,

(2) GLOBAL\_AUDIO\_LOCKOUT\_STATUS, (3) ZONE\_SEAT\_AUDIO\_LOCKOUT\_STATUS,

(4) SEAT\_CLIMATE\_LOCKOUT\_STATUS, (5) SEAT\_MOVEMENT\_LOCKOUT\_STATUS,

(6) SEAT\_MASSAGE\_LOCKOUT\_STATUS, (7) DIGITAL\_SHADE\_LOCKOUT\_STATUS,

(8) ALL\_FUNCTIONS\_LOCKOUT\_STATUS. Each sub signal indicate the lockout functionality status via LOCKED or NOT\_LOCKED

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of LockoutFunctionality

VentDistribuition

"VentDistribuition" signal to select ventilation location via "FRONT", "FEET", or "BOTH"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **FRONT** |  |
| **FEET** |  |
| **BOTH** |  |
| **Unit** | |  |

Table: Encoding Details of VentDistribuition

DigitalShadeSettings

"DigitalShadeSettings" to identify URC user's digital shade settings via DIGTAL\_SETTING\_1, DIGTAL\_SETTING\_2

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **DIGTAL\_SETTING\_1** |  |
| **DIGITAL\_SETTING\_2** |  |
| **Unit** | |  |

Table: Encoding Details of DigitalShadeSettings

LEFT\_FIRST\_ROW

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **INHIBIT\_GLOBAL\_AUDIO** |  |
| **INHIBIT\_INERIOR\_LIGHTING** |  |
| **INHIBIT\_ DIGITAL\_SHADE** |  |
| **Unit** | |  |

Table: Encoding Details of LEFT\_FIRST\_ROW

SeatMovementSettings

"SeatPositionSettings" signal identify seat position settings; seat cushion movement via "SeatCushionMovement ", seat back movement via "SeatBackMovement ", head rest movement via "HeadRestMovement", bladders via "Bladders"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of SeatMovementSettings

ZONE\_SEAT\_AUDIO

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **AVAILABLE** |  |
| **UNAVAILABLE** |  |
| **Unit** | |  |

Table: Encoding Details of ZONE\_SEAT\_AUDIO

Seat\_Massage\_Request

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of Seat\_Massage\_Request

INTERIOR\_LIGHTING

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **AVAILABLE** |  |
| **UNAVAILABLE** |  |
| **Unit** | |  |

Table: Encoding Details of INTERIOR\_LIGHTING

RRClimateButtonStatus

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of RRClimateButtonStatus

ConnectedSeaID

"ConnectedSeaID" to identify URC user's seat location via LEFT\_FIRST\_ROW, RIGHT\_FIRST\_ROW, LEFT\_SECOND\_ROW, RIGHT\_SECOND\_ROW, LEFT\_THIRD\_ROW, RIGHT\_THIRD\_ROW, NONE

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **LEFT\_FIRST\_ROW** |  |
| **RIGHT\_FIRST\_ROW** |  |
| **LEFT\_SECOND\_ROW** |  |
| **RIGHT\_SECOND\_ROW** |  |
| **LEFT\_THIRD\_ROW** |  |
| **RIGHT\_THIRD\_ROW** |  |
| **NOSEATCONNECTED** |  |
| **Unit** | |  |

Table: Encoding Details of ConnectedSeaID

SEAT\_MASSAGE\_LOCKOUT\_STATUS

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **LOCKED** |  |
| **NOT\_LOCKED** |  |
| **Unit** | |  |

Table: Encoding Details of SEAT\_MASSAGE\_LOCKOUT\_STATUS

LightIntensity

“LightIntensity” signal define interior light color intensity

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of LightIntensity

ClimateSettings

"ClimateSettings" signal control zone climate via "DesiredFanSpeed", "DesiredTemperature","TempUnit", and "VentDistribuition"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of ClimateSettings

LightingStatus

"LightingStatus" signal display lighting status via "UPDATE\_SUCCESSFUL","LIGHTING\_SCHEME\_UPDATE\_FAILED","LIGHTING\_COLOR\_UPDATE\_FAILED","LIGHTING\_INTENSITY\_UPDATE\_FAILED","LIGHTING\_ON\_UPDATE\_FAILED","LIGHTING\_OFF\_UPDATE\_FAILED","NOT\_APPLICABLE"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **UPDATE\_SUCCESSFUL** |  |
| **LIGHTING\_SCHEME\_UPDATE\_FAILED** |  |
| **LIGHTING\_COLOR\_UPDATE\_FAILED** |  |
| **LIGHTING\_INTENSITY\_UPDATE\_FAILED** |  |
| **LIGHTING\_ON\_UPDATE\_FAILED** |  |
| **LIGHTING\_OFF\_UPDATE\_FAILED** |  |
| **NOT\_APPLICABLE** |  |
| **Unit** | |  |

Table: Encoding Details of LightingStatus

SeatCushionMovement

“SeatCushionMovement ” signal define seat cushion movement settings via CUSHION\_TRACK\_FORE,

CUSHION\_TRACK\_AFT, CUSHION\_TRACK\_UP, CUSHION\_TRACK\_DOWN, CUSHION\_TRACK\_TILT\_FORWARD, CUSHION\_TRACK\_TIL\_BACKWARD, CUSHION\_EXTENSION\_LEFT, CUSHION\_EXTENSION\_RIGHT, CALF\_RAISE\_UP, CALF\_RAISE\_DOWN

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **CUSHION\_TRACK\_FORE** |  |
| **CUSHION\_TRACK\_AFT** |  |
| **CUSHION\_TRACK\_UP** |  |
| **CUSHION\_TRACK\_DOWN** |  |
| **CUSHION\_TRACK\_TILT\_FORWARD** |  |
| **CUSHION\_TRACK\_TIL\_BACKWARD** |  |
| **CUSHION\_EXTENSION\_LEFT** |  |
| **CUSHION\_EXTENSION\_RIGHT** |  |
| **CALF\_RAISE\_UP** |  |
| **CALF\_RAISE\_DOWN** |  |
| **Unit** | |  |

Table: Encoding Details of SeatCushionMovement

UserConfigurationSettings

"UserConfigurationSettings" signal identify user configuration settings via Sub-Signals; "LightCommand","AudioSettings","ClimateSettings","SeatSettings"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of UserConfigurationSettings

SeatAudioActions

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **RECEIVE\_AUDIO\_SHARE** |  |
| **ENABLE\_DO \_NOT\_DISTURB** |  |
| **DISABLE\_DO\_NOT\_DISTURB** |  |
| **INITIATE\_IN\_CAR\_COMMUNICATION** |  |
| **RECEIVE\_IN\_CAR\_COMMUNICATION** |  |
| **Unit** | |  |

Table: Encoding Details of SeatAudioActions

Intensity

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **OFF** |  |
| **LOW** |  |
| **MEDIUM** |  |
| **HIGH** |  |
| **Unit** | |  |

Table: Encoding Details of Intensity

SeatHeat

“SeatHeat” signal define seat heat settings via NONE, LOW, MEDIUM, HIGH

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **NONE** |  |
| **LOW** |  |
| **MEDIUM** |  |
| **HIGH** |  |
| **Unit** | |  |

Table: Encoding Details of SeatHeat

SeatMassageSettings

"SeatMassage" signal identify; seat massage pattern via Sub-signal "Pattern" that specify pattern levels through "PATTERN1, PATTERN2, PATTERN 3“. Also, it identifies seat massage intensity level via sub-signal "Intensity" that specify the intensity level through "OFF, LOW, MEDIUM, HIGH"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of SeatMassageSettings

SEAT\_CLIMATE\_LOCKOUT\_STATUS

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **LOCKED** |  |
| **NOT\_LOCKED** |  |
| **Unit** | |  |

Table: Encoding Details of SEAT\_CLIMATE\_LOCKOUT\_STATUS

LightONOFF

“LightONOFF” signal define Light on off status via ON and OFF

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **OFF** |  |
| **ON** |  |
| **Unit** | |  |

Table: Encoding Details of LightONOFF

Audio\_Status

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Request\_Volume\_Change\_Status** |  |
| **Request\_Source\_Change\_Status** |  |
| **Request\_Seek\_Change\_Status** |  |
| **Request\_Audio\_Share\_Status** |  |
| **Request\_ICC\_Status** |  |
| **Request\_DnD\_Status** |  |
| **Request\_Zone\_Volume\_Change\_Status** |  |
| **Unit** | |  |

Table: Encoding Details of Audio\_Status

UserConnectionRequestAction

UserConnectionRequestAction” signal for the URC user to request connection to URC

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of UserConnectionRequestAction

AudioSettingsStatus

"AudioSettingsStatus" signal display audio settings status via "UPDATE\_SUCCESSFUL","AUDIO\_LEVEL\_PERCENT\_FAILED","AUDIO\_SETTINGS\_FAILED","NOT\_APPLICABLE"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **UPDATE\_SUCCESSFUL** |  |
| **AUDIO\_LEVEL\_PERCENT\_FAILED** |  |
| **AUDIO\_SETTINGS\_FAILED** |  |
| **NOT\_APPLICABLE** |  |
| **Unit** | |  |

Table: Encoding Details of AudioSettingsStatus

INTERIOR\_LIGHTING\_LOCKOUT\_STATUS

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **LOCKED** |  |
| **NOT\_LOCKED** |  |
| **Unit** | |  |

Table: Encoding Details of INTERIOR\_LIGHTING\_LOCKOUT\_STATUS

Climate\_Request

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Request\_Temp\_Quad1** |  |
| **Request\_Temp\_Quad2** |  |
| **Request\_Temp\_Quad3** |  |
| **Request\_Power\_On/Off\_Quad3** |  |
| **Request\_Auto\_On/Off\_Quad3** |  |
| **Request\_Fan\_Speed\_Quad3** |  |
| **Request\_Vent\_Mode\_Quad3** |  |
| **Request\_Power\_On/Off\_Quad4** |  |
| **Request\_Auto\_On/Off\_Quad4** |  |
| **Request\_Fan\_Speed\_Quad4** |  |
| **Request\_Vent\_Mode\_Quad4** |  |
| **Request\_Power\_On/Off\_Quad4** |  |
| **Request\_Auto\_On/Off\_Quad4** |  |
| **Unit** | |  |

Table: Encoding Details of Climate\_Request

LightColor

“LightColor” signal define interior light color settings via -RED : Integer, -GREEN : Integer, -BLUE : Integer. 0-255

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **RED** | 0-255 |
| **GREEN** | 0-255 |
| **BLUE** | 0-255 |
| **Unit** | |  |

Table: Encoding Details of LightColor

Request\_Digital\_Shade

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of Request\_Digital\_Shade

RIGHT\_SECOND\_ROW

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **INHIBIT\_CLIMATE** |  |
| **INHIBIT\_GLOBAL\_AUDIO** |  |
| **INHIBIT\_INERIOR\_LIGHTING** |  |
| **INHIBIT\_DIGITAL\_SHADE** |  |
| **Unit** | |  |

Table: Encoding Details of RIGHT\_SECOND\_ROW

UserAcceptanceClassification

“UserAcceptanceClassification” signal define user acceptance classification

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of UserAcceptanceClassification

SeatSettings

"SeatSettings" signal defines seat settings via three sub signals

1) "SeatMassage" specified by "Pattern" levels (PATTERN1, PATTERN2, PATTERN 3) and "Intensity" measures of (OFF, LOW, MEDIUM, HIGH)

2) "SeatPositionSettings" specified by seat cushion movement via "SeatCushionMovement ",seat back movement via "SeatBackMovement", head rest movement via "HeadRestMovement", and bladders via "Bladders"

3) "SeatClimate" specified by seat venting via "SeatVenting", and seat heating via "SeatHeating"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of SeatSettings

AvailableFunctionality

“AvailableFunctionality” signal contains 8 sub signal (1) INTERIOR\_LIGHTING, (2) GLOBAL\_AUDIO,

(3) ZONE\_SEAT\_AUDIO, (4) SEAT\_CLIMATE, (5) SEAT\_MOVEMENT, (6) SEAT\_MASSAGE, (7) DIGITAL\_SHADE, (8) ALL\_FUNCTIONS. Each sub signal indicate the available functionality status via AVAILABLE or UNAVAILABLE

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of AvailableFunctionality

AudioType

"AudioType" signal specify audio type via “GLOBAL\_AUDIO“ and “ZONE\_AUDIO”

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **GLOBAL\_AUDIO** |  |
| **ZONE\_AUDIO** |  |
| **Unit** | |  |

Table: Encoding Details of AudioType

DriverInhibitFunctionality

"InhibitFunctionality" signal is to inhibit URC functionality via “INHIBIT\_CLIMATE","INHIBIT\_GLOBAL\_AUDIO","INHIBIT\_ZONE\_SEAT\_AUDIO","NONE“,”INHIBIT\_URC”

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **INHIBIT\_CLIMATE** |  |
| **INHIBIT\_AUDIO** |  |
| **NONE** |  |
| **INHIBIT\_URC** |  |
| **Unit** | |  |

Table: Encoding Details of DriverInhibitFunctionality

TempUnit

"TempUnit" signal to select temperature unit either F or C via "F" or "C"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **F** |  |
| **C** |  |
| **Unit** | |  |

Table: Encoding Details of TempUnit

UserLeftVehicle

“UserLeftVehicle” signal to identify if the URC user left vehicle via USER\_LEFT\_VEHICLE or did not leave vehicle via USER\_IN\_VEHICLE

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **USER\_LEFT\_VEHICLE** |  |
| **USER\_IN\_VEHICLE** |  |
| **Unit** | |  |

Table: Encoding Details of UserLeftVehicle

AudioCommand

“AudioCommand” a signal contains 2 sub signals (1) GlobalAudioSettings to define global audio sound level via GlobalAudioLevelPercent signal and audio source via AudioSource signal and audio action via AudioAction signal

(2) SeatAudioSettings to define seat audio sound level via SeatAudioLevelPercent signal

and seat audio actions via SeatAudioActions signal

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of AudioCommand

GLOBAL\_AUDIO

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **AVAILABLE** |  |
| **UNAVAILABLE** |  |
| **Unit** | |  |

Table: Encoding Details of GLOBAL\_AUDIO

FunctionType

"FunctionType" signal defines functions type that URC controls Via; "DIGITAL\_SHADE", "ZONE\_SEAT\_AUDIO"," ZONE\_CLIMATE"," SEAT\_CLIMATE"," INTERIOR\_LIGHTING"," GLOBAL\_AUDIO"," SEAT\_POSITION"," SEAT\_MASSAGE"," ALL\_FUNCTIONS"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **INTERIOR\_LIGHTING** |  |
| **GLOBAL\_AUDIO** |  |
| **SEAT\_CLIMATE** |  |
| **ZONE\_CLIMATE** |  |
| **ZONE\_SEAT\_AUDIO** |  |
| **DIGITAL\_SHADE** |  |
| **SEAT\_MASSAGE** |  |
| **ALL\_FUNCTIONS** |  |
| **SEAT\_MOVEMENT** |  |
| **AUDIO** |  |
| **SEAT\_POSITION** |  |
| **Unit** | |  |

Table: Encoding Details of FunctionType

FirstRowConnectionInvitationAction

“FirstRowConnectionInvitationAction” signal to indicate 1R Passenger broadcast vehicle connection invitation to URC

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of FirstRowConnectionInvitationAction

ALL\_FUNCTIONS

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **AVAILABLE** |  |
| **UNAVAILABLE** |  |
| **Unit** | |  |

Table: Encoding Details of ALL\_FUNCTIONS

Rejuvenate

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of Rejuvenate

InhibitedSeatID

“InhibitedSeatID” signal to identify 2R and 3R URC user’s inhibited seat identification via; LEFT\_SECOND\_ROW, RIGHT\_SECOND\_ROW, LEFT\_THIRD\_ROW, RIGHT\_THIRD\_ROW, NONE

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **LEFT\_SECOND\_ROW** |  |
| **RIGHT\_SECOND\_ROW** |  |
| **LEFT\_THIRD\_ROW** |  |
| **RIGHT\_THIRD\_ROW** |  |
| **NONE** |  |
| **Unit** | |  |

Table: Encoding Details of InhibitedSeatID

Seat\_Movement\_Status

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_horizontal\_Status** |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_vertical\_Status** |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_tilt\_Status** |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_recliner\_Status** |  |
| **MoveSeatRqst-URC-DRIVER\_seat\_2way\_lumbar\_Status** |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_headrest\_horizontal\_Status** |  |
| **MoveSeatRqst\_URC,DRIVER,seat\_headrest\_vertical\_Status** |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_left\_thigh\_ext\_Status** |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_right\_thigh\_ext\_Status** |  |
| **MoveSeatRqstURC\_DRIVER\_seat\_thoracic\_pivot\_Status** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_horizontal\_Status** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_vertical\_Status** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_tilt\_Status** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_recliner\_Status** |  |
| **MoveSeatRqst\_URC-1R PASS-seat\_2way\_lumbar\_Status** |  |
| **MoveSeatRqst\_URC\_1R PASS\_eat\_headrest\_horizontal\_Status** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_headrest\_vertical\_Status** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_left\_thigh\_ext\_Status** |  |
| **MoveSeatRqst\_URC\_1R PASS-seat\_right\_thigh\_ext\_Status** |  |
| **MoveSeatRqst\_URC\_1R PASS\_eat\_thoracic\_pivot\_Status** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_calf\_raise\_Status** |  |
| **MoveSeatRqst\_URC\_2R LH-seat\_horizontal\_Status** |  |
| **MoveSeatRqst\_URC-2R LH\_seat\_recliner\_Status** |  |
| **MoveSeatRqst\_URC\_2R LH,seat\_calf\_raise\_Status** |  |
| **MoveSeatRqst\_URC-2R LH\_seat\_2way\_lumbar\_Status** |  |
| **MoveSeatRqst\_URC\_2R RH\_seat\_horizontal\_Status** |  |
| **MoveSeatRqst\_URC\_2R RH\_seat\_recliner\_Status** |  |
| **MoveSeatRqst\_URC\_2R RH\_seat\_calf\_raise\_Status** |  |
| **MoveSeatRqst\_URC\_2R RH\_seat\_2way\_lumbar\_Status** |  |
| **Unit** | |  |

Table: Encoding Details of Seat\_Movement\_Status

Seat\_Movement\_Request

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **MoveSeatRqst\_URC,DRIVER,seat\_horizontal** |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_vertical** |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_tilt** |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_recliner** |  |
| **MoveSeatRqst-URC-DRIVER\_seat\_2way\_lumbar** |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_headrest\_horizontal** |  |
| **MoveSeatRqst\_URC,DRIVER,seat\_headrest\_vertical** |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_left\_thigh\_ext** |  |
| **MoveSeatRqst\_URC\_DRIVER\_seat\_right\_thigh\_ext** |  |
| **MoveSeatRqstURC\_DRIVER\_seat\_thoracic\_pivot** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_horizontal** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_vertical** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_tilt** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_recliner** |  |
| **MoveSeatRqst\_URC-1R PASS-seat\_2way\_lumbar** |  |
| **MoveSeatRqst\_URC\_1R PASS\_eat\_headrest\_horizontal** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_headrest\_vertical** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_left\_thigh\_ext** |  |
| **MoveSeatRqst\_URC\_1R PASS-seat\_right\_thigh\_ext** |  |
| **MoveSeatRqst\_URC\_1R PASS\_eat\_thoracic\_pivot** |  |
| **MoveSeatRqst\_URC\_1R PASS\_seat\_calf\_raise** |  |
| **MoveSeatRqst\_URC\_2R LH-seat\_horizontal** |  |
| **MoveSeatRqst\_URC-2R LH\_seat\_recliner** |  |
| **MoveSeatRqst\_URC\_2R LH,seat\_calf\_raise** |  |
| **MoveSeatRqst\_URC-2R LH\_seat\_2way\_lumbar** |  |
| **MoveSeatRqst\_URC\_2R RH\_seat\_horizontal** |  |
| **MoveSeatRqst\_URC\_2R RH\_seat\_recliner** |  |
| **MoveSeatRqst\_URC\_2R RH\_seat\_calf\_raise** |  |
| **MoveSeatRqst\_URC\_2R RH\_seat\_2way\_lumbar** |  |
| **Unit** | |  |

Table: Encoding Details of Seat\_Movement\_Request

SeatClimateSettings

"SeatHeating" signal identify seat climate as seat ventilation via "SeatVenting" and seat heat via "SeatHeating"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of SeatClimateSettings

DriverAcceptanceClassification

DriverAcceptanceClassification” signal for the 1R to Check on the URC user Connection Classification as temporary or permanent

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of DriverAcceptanceClassification

DIGITAL\_SHADE

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **AVAILABLE** |  |
| **UNAVAILABLE** |  |
| **Unit** | |  |

Table: Encoding Details of DIGITAL\_SHADE

ZONE\_SEAT\_AUDIO\_LOCKOUT\_STATUS

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **LOCKED** |  |
| **NOT\_LOCKED** |  |
| **Unit** | |  |

Table: Encoding Details of ZONE\_SEAT\_AUDIO\_LOCKOUT\_STATUS

InteriorLightRequest

"InteriorLightCommand" signal selects sub-signals; "LightScheme","LightColor","LightIntensity","LightCommand","LightType"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **LightScheme** |  |
| **LightColor** |  |
| **LightIntensity** |  |
| **LightCommand** |  |
| **Unit** | |  |

Table: Encoding Details of InteriorLightRequest

FRClimateButtonStatus

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **None\_Pressed** |  |
| **LHS\_Temp\_Inc\_Pressed** |  |
| **LHS\_Temp\_Dec\_Pressed** |  |
| **RHS\_Temp\_Inc\_Pressed** |  |
| **RHS\_Temp\_Dec\_Pressed** |  |
| **Unit** | |  |

Table: Encoding Details of FRClimateButtonStatus

VehicleConnectionList

"VehicleConnectionList" a signal that defines if the URC user is

1) a permanent user connected the first time or second, third time or

P\_CONNECTION\_1

P\_CONNECTION\_2

P\_CONNECTION\_3

2) a temporary user connected the first time or second,third time or

T\_CONNECTION\_1

T\_CONNECTION\_2

T\_CONNECTION\_3

3) a new user connecting

NEW

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **P\_CONNECTION\_1** |  |
| **P\_CONNECTION\_2** |  |
| **P\_CONNECTION\_3** |  |
| **T\_CONNECTION\_1** |  |
| **T\_CONNECTION\_2** |  |
| **T\_CONNECTION\_3** |  |
| **Unit** | |  |

Table: Encoding Details of VehicleConnectionList

SeatBladders

“SeatBladders” signal define seat bladders settings via BACK\_BOLSTER\_BLADDERS,CUSHION\_BOLSTER\_BLADDERS

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **BACK\_BOLSTER\_BLADDERS** |  |
| **CUSHION\_BOLSTER\_BLADDERS** |  |
| **Unit** | |  |

Table: Encoding Details of SeatBladders

HeadRestMovement

“HeadRestMovement” signal define head rest movement settings via HEAD\_REST\_FORE, HEAD\_REST\_AFT, HEAD\_RES\_ UP, HEAD\_REST\_DOWN

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **HEAD\_REST\_FORE** |  |
| **HEAD\_REST\_AFT** |  |
| **HEAD\_RES\_ UP** |  |
| **HEAD\_REST\_DOWN** |  |
| **Unit** | |  |

Table: Encoding Details of HeadRestMovement

UserLeaveVehicle

"UserLeaveVehicle" to define URC user leave vehicle

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of UserLeaveVehicle

RIGHT\_FIRST\_ROW

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **INHIBIT\_CLIMATE** |  |
| **INHIBIT\_GLOBAL\_AUDIO** |  |
| **INHIBIT\_INERIOR\_LIGHTING** |  |
| **INHIBIT\_DIGITAL\_SHADE** |  |
| **Unit** | |  |

Table: Encoding Details of RIGHT\_FIRST\_ROW

DesiredFanSpeed

"DesiredFanSpeed" signal determine the incremental fan speed from "ZERO\_PRCNT", "TEN\_PRCNT", "TWENTY\_PRCNT" to "HUNDRED\_PRCNT"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **ZERO\_PRCNT** |  |
| **TEN\_PRCNT** |  |
| **TWENTY\_PRCNT** |  |
| **THIRTY\_PRCNT** |  |
| **FOURTY\_PRCNT** |  |
| **FIFTY\_PRCNT** |  |
| **SIXTY\_PRCNT** |  |
| **SEVENTY\_PRCNT** |  |
| **EIGHTY\_PRCNT** |  |
| **NINTY\_PRCNT** |  |
| **HUNDRED\_PRCNT** |  |
| **Unit** | |  |

Table: Encoding Details of DesiredFanSpeed

RIGHT\_THIRD\_ROW

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **INHIBIT\_CLIMATE** |  |
| **INHIBIT\_GLOBAL\_AUDIO** |  |
| **INHIBIT\_INERIOR\_LIGHTING** |  |
| **INHIBIT\_DIGITAL\_SHADE** |  |
| **Unit** | |  |

Table: Encoding Details of RIGHT\_THIRD\_ROW

ALL\_FUNCTIONS\_LOCKOUT\_STATUS

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **LOCKED** |  |
| **NOT\_LOCKED** |  |
| **Unit** | |  |

Table: Encoding Details of ALL\_FUNCTIONS\_LOCKOUT\_STATUS

RejuvenateStatus

“RejuvenateStatus” signal to identify Rejuvenate status as active via ACTIVE or inactive via INACTIVE

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **ACTIVE** |  |
| **INACTIVE** |  |
| **Unit** | |  |

Table: Encoding Details of RejuvenateStatus

SEAT\_CLIMATE

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **AVAILABLE** |  |
| **UNAVAILABLE** |  |
| **Unit** | |  |

Table: Encoding Details of SEAT\_CLIMATE

GlobalAudioSettings

"AudioSettings" signal defines audio settings via "AudioLevelPercent" and "AudioONOFF"

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of GlobalAudioSettings

RejuvenateInhibit

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of RejuvenateInhibit

URCDisplayStatus

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **Unit** | |  |

Table: Encoding Details of URCDisplayStatus

LEFT\_THIRD\_ROW

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **INHIBIT\_CLIMATE** |  |
| **INHIBIT\_GLOBAL\_AUDIO** |  |
| **INHIBIT\_INERIOR\_LIGHTING** |  |
| **INHIBIT\_DIGITAL\_SHADE** |  |
| **Unit** | |  |

Table: Encoding Details of LEFT\_THIRD\_ROW

Request Climate Functionality

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Yes** |  |
| **No** |  |
| **Unit** | |  |

Table: Encoding Details of Request Climate Functionality

VehicleConnectionStatus

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of VehicleConnectionStatus

Light\_Status

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Request\_Light\_On/Off Status** |  |
| **Request\_LightAmbColor\_No\_Rq Status** |  |
| **Request\_LightAmbIntsty\_No\_Rq Status** |  |
| **Unit** | |  |

Table: Encoding Details of Light\_Status

ClimateType

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **TEMPERATURE** |  |
| **FANSPEED** |  |
| **VENTDISTRIBUITION** |  |
| **Unit** | |  |

Table: Encoding Details of ClimateType

Seat\_Massage\_Status

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **Unit** | |  |

Table: Encoding Details of Seat\_Massage\_Status

SeatAudioLevelPercent

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **0\_PERCENT** |  |
| **10\_PERCENT** |  |
| **20\_PERCENT** |  |
| **30\_PERCENT** |  |
| **40\_PERCENT** |  |
| **50\_PERCENT** |  |
| **60\_PERCENT** |  |
| **70\_PERCENT** |  |
| **80\_PERCENT** |  |
| **90\_PERCENT** |  |
| **100\_PERCENT** |  |
| **Unit** | |  |

Table: Encoding Details of SeatAudioLevelPercent

InhibitFunctionsAffectRejuvenate

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **INHIBIT\_GLOBAL\_AUDIO** |  |
| **INHIBIT\_INERIOR\_LIGHTING** |  |
| **INHIBIT\_DIGITAL\_SHADE** |  |
| **INHIBIT\_RIGHT\_FIRST\_ROW\_CLIMATE** |  |
| **INHIBIT\_LEFT\_SECOND\_ROW\_CLIMATE** |  |
| **INHIBIT\_LEIFT\_THIRD\_ROW\_CLIMATE** |  |
| **INHIBIT\_RIGHT\_SECOND\_\_ROW\_CLIMATE** |  |
| **INHIBIT\_RIGHT\_THRID\_ROW\_CLIMATE** |  |
| **Unit** | |  |

Table: Encoding Details of InhibitFunctionsAffectRejuvenate

GlobalAudioLevelPercent

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **10\_PERCENT** |  |
| **20\_PERCENT** |  |
| **30\_PERCENT** |  |
| **40\_PERCENT** |  |
| **50\_PERCENT** |  |
| **60\_PERCENT** |  |
| **70\_PERCENT** |  |
| **80\_PERCENT** |  |
| **90\_PERCENT** |  |
| **100\_PERCENT** |  |
| **0\_PERCENT** |  |
| **Unit** | |  |

Table: Encoding Details of GlobalAudioLevelPercent

GLOBAL\_AUDIO\_LOCKOUT\_STATUS

**Note:** An encoding is either discrete or continuous. Delete those fields, which are not needed.

|  |  |  |
| --- | --- | --- |
| **Value**  (Continuous Encoding) | Min Value |  |
| Max Value |  |
| Resolution |  |
| Offset |  |
| **Value**  (Discrete  Encoding) |  |  |
| **LOCKED** |  |
| **NOT\_LOCKED** |  |
| **Unit** | |  |

Table: Encoding Details of GLOBAL\_AUDIO\_LOCKOUT\_STATUS

Document ends here.