|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | |  |
|  |  | | |  |
|  | Function Group Spec  HMI System  <<Logical>> (Allocated) | | |  |
|  |  |
|  |  |
|  |  | | |  |
| Document Type | **Function Specification** | | |  |
| Template Version | **6.1a** | | |  |
| SysML Report Template Version | **M (03/09/2021)** | | |  |
| Document ID | **functionspec\_sysmlreporttemplate** | | |  |
| Document Location |  | | |  |
| Document Owner |  | | |  |
| Document Revision | **FGS1** | | |  |
| Document Status | **Draft** | | |  |
| Date Issued | **2021/03/04** | | |  |
| Date Revised | **2021/03/04** | | |  |
| Document Classification | GIS1 Item Number: | **27.60/35** | |  |
| GIS2 Classification: | **Confidential** | |
|  | | | | |
|  | | | | |
| Document Approval | | | | |
| Name | Role | | Email Confirmation | Date |
|  |  | |  |  |
|  |  | |  |  |

Auto-Generated by MagicDraw

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.**

**Copyright, Ó 2021 Ford Motor Company**

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FORD MOTOR COMPANY reserves the right to protect this work as an unpublished copyrighted work in the event of an inadvertent or deliberate unauthorized publication.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 to any duly authorized representative of Ford Motor Company.

# Contents

1.1.1 Decomposition of 4

Disclaimer 16

Contents 17

2 Introduction 19

2.1 Document Purpose 19

2.2 Document Scope 19

2.3 Document Audience 19

2.3.1 Stakeholder List 19

2.4 Document Organization 19

2.4.1 Document Context 19

2.4.2 Document Structure 19

2.5 Document Conventions 20

2.5.1 Requirements Templates 20

3 Function Group Description 21

4 Functional Architecture 22

4.1 Description 22

4.2 Function List 22

4.3 Signal List 22

5 Function Specifications 23

5.1  23

5.1.1 Function Overview 23

5.1.2 Function Scope 25

5.1.3 Function Interfaces 26

5.1.4 Function Modeling 28

5.1.5 Function Requirements 28

6 Open Concerns 32

7 Revision History 33

8 Appendix 34

8.1 Data Dictionary 34

8.1.1 Logical Signals 34

8.1.2 Logical Parameters 34

8.1.3 Encoding Types 34

8.2 Glossary 34

8.2.1 Definitions 34

8.2.2 Abbreviations 35

**List of Figures**

Figure 1 Activity diagram “” depicting black box behavior collaboration 22

Figure 2: Activity Diagram of “” calling “” 26

Figure 3: 28

**List of Tables**

Table 1: - Decomposition Table 5

Table 2: Functions described in this specification 19

Table 3: List of Logical Functions 22

Table 4: Ford internal Documents 25

Table 5: Ford internal Documents *(not specified in model)* 25

Table 6: External documents and publications 25

Table 7: External documents and publications *(not specified in model)* 25

Table 8: Open Concerns *(Not supported by MagicDraw report generation.)* 32

Table 9: Definitions used in this document 35

Table 10: Abbreviations used in this document 35

# Introduction

## Document Purpose

The Function (Group) Specification (FS) specifies an individual function / a group of functions.

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

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Function ID** | **Function Name** | **Owner** | **Reference** |

**Table 2: Functions described in this specification**

## Document Audience

The FS is authored by the owners of the individual functions. All Stakeholders, i.e., all people who have a valid interest in the functions and their behavior should read and, if possible, review the FS. It needs to be guaranteed, that all stakeholders have access to the currently valid version of the FS.

### Stakeholder List

For the latest list of the feature stakeholder and their roles & responsibilities refer to <Put VSEM Link here>.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **CDSID** | **Contact Info** | **Role** | **Stakeholder Group** |
| Vikaram Gokhale | vgokhale |  | Feature Delivery Supervisor |  |
| Landie Berry | cberry37 |  | HMI UX |  |
| William Johnston | wjohnst2 |  | Supervisor Occupant Features |  |
| Robert Paquette | rpaquet2 |  | Infotainment Systems Engr |  |
| Dinh Tran | dtran39 |  | Infotainment Systems Engr |  |
| Jason Richardson | jricha41 |  | Climate Core Engr | VESC |
| Michelle Churchill | mchurc35 |  | D & R Engineer |  |
| Andrew Zourob | azourob |  | Climate Control Core Engr | VESC |
| Ken Cunningham | kcunni16 |  | FuSa Coach |  |
| George Smith | GSMIT560 |  | Feature Owner | Interior Cockpit and Trim Core Engineerint |
| Steven Jacobson | sjcobse |  | Feature Champion |  |
| Christopher Van Auken | cvanauke |  | RCCM Engineer | VESC |
| Andres Portillo Reyes | aportil9 |  | Climate Hardware Engr | VESC |
| Aamir Pasha | apasha |  | Feature Delivery Engineer |  |
| Martin Lindell | mlindel4 |  | Climate CFD |  |
| Mitali Chakrabarti | mchakrab |  | Climate Features Supervisor |  |
| Neelima Majjiga | nmajjiga |  | Infotainment Systems Engr |  |
| Julee Harlow | jharlow6 |  | HMI Supervisor |  |
| Jamie Liao | jliao |  | Climate Hardware Engr | VESC |
| Andrew Zourob | azourob |  | Climate Control Core Engr |  |
| Laura Check | LBUREK |  | SYNC3 Supervisor |  |
| Jorge Aaujo | jarauj59 |  | Register Engineer |  |
| Joe Qussar | jqussar |  | FuSa Engr |  |
| Demetrius Johnson-Gault | djohn840 |  | HMI Core Engr |  |
| Joshua Cerda | jcerda7 |  | Modeling Team Member |  |
| Greg Kopp | gkopp6 |  | HMI Engineer |  |
| Martin Imhof | mimhof4 |  | Model Architect |  |
| Tao Zhou | tzhou32 |  | Climate Core Engr | VESC |
| Fernando OVando | joveaneo |  | Register Engineer |  |
| Cindy Rutyna | crutyna |  | Supervisor Climate E/E | VESC |

## 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 FS relates to other Ford Requirements Documents and Specifications.

### Document Structure

The structure of this document is explained below:

**Section 1** – Introduction how to use this document including responsibilities and requisite documents. Explains the tterminology. Gives a clarification of the definitions, concepts and abbreviations used in the document.

**Section 2** – Function Group Description. Gives an overview and the purpose of the function group.

**Section 3** – Functional Architecture: Specifies the overall functional architecture of the function group

**Section 4** – Function Specifications: Specifies the logical functions of the function group in detail

**Section 5** – List of Open Concerns

**Section 6** Revision history including a list of new or modified requirements. The requirements in this document are tagged, and this section contains different types of tables listing all, new, or changed requirements by their title and page no.

**Section 7** – 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 a FS 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\_FNC\_LockArbitrator\_00004* This is the fourth requirement on function level for the function Lock Arbitrator.

#### Requirements Attributes

The templates provided by *Specification\_Macros.dotm* define a list of attributes for each requirement. This helps to classify the requirement. The attributes are explained at [RE Wiki - Requirements Attributes](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes?src=contextnavpagetreemode).

## References

### Ford Documents

List here all Ford internal documents, which are directly related to the feature.

| **Reference** | **Title** | **Doc. ID** | **Document Location** | **Revision** |
| --- | --- | --- | --- | --- |
| Ford GIS Standard | Ford GIS Standard |  |  |  |

Table 2: Ford Documents

### External Documents and Publications

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

| **Reference** | **Document / Publication** | **Document Location** |
| --- | --- | --- |
| IEEE Std 1012-2004 IEEE Standard for Software Verification and Validation |  |  |
| ISO/IEC 19500-2:2003 | Information technology -- Open Distributed Processing -- Part 2 |  |
| UML Testing Profile (UTP), v1.2 |  |  |
| Wikipedia |  |  |

Table 3: External Documents and Publications

## Glossary

### Definitions

| **Definition** | **Description** |
| --- | --- |
| APIM | Auxillary Protocol Interface Module (Sync Gen 4.2) with Cluster |
| BeforeClosedPosition | The last stationary positions that the registers were in before the registers were closed. To be used to return the registers back to their previous position when the registers are requested to open by the User. |
| calibrate | A process that will provide future accurate positioning for the registers. The process entails moving the registers to the end points and recording the feedback from the sensors. This feedback is used to recalibrate the future voltage commands to move the registers. |
| CIS | Controls Interface Specification |
| High speed | Approximately more than 52 mph (83 kph) |
| HOME | Value that denotes the Home position the registers will move to when the car is turned off. |
| LastPosition | Denotes the last stationary position that the registers were in. For example the position before the vehicle was turned OFF, before a Cycling option was selected, or before another feature requested to move the registers. |
| Low speed | Approximately 12 to 36 mph (19 to 58 kph ) |
| MaxPositionError | The maximum degree error allowed for a register position before the position is considered incorrect. |
| Medium speed | Approximately 36 mph to 52 mph (58 to 83 kph) |
| RCCM | Remote Climate Control Module |
| Specified point to point rate | Rate at which register vanes move. |
| StaticSelection | Register movement selection that is not a dynamic option |
| status of EM registers |  |
| term | A representation of a Concept expressed in Natural Language. In the vocabulary of a Domain of Discourse a term enables common understanding of domain concepts. |
| term glossary | A term glossary is a table of agreed upon definitions for terms used in project development that may provide clarity or avoid confusion to stakeholders. |
| Threshold | The minimum change that the register will move to. |
| TLA | Three Letter Acronym |
| Very Low Speed | Approximately 0 to 12 mph (0 to 19 kph) |

Table 4: Definitions used in this document

### Abbreviations

| **Abbr.** | **Meaning** | **Description** |
| --- | --- | --- |
| ATLA | Another Three Letter Acronym |  |

Table 5: Abbreviations used in this document

# Function Group Description

This Function Group consists of all functions allocated to 44569178.jpg **HMI System** <<Logical>> including all functions in their corresponding call trees.

Description of HMI System:

EM Registers communicates with Rejuvenate through the HMI System because Rejuvenate is part of the HMI system. HMI system stand in place of the User.

# Functional Decomposition and Architecture

## Description

## Function List

|  |  |  |  |
| --- | --- | --- | --- |
| **Function ID** | **Function Name** | **Function Description** | **ASIL** |
|  | -775985605.jpg [Detect User Input](#_c89354962997b027dab358481fb9d137) <<Subsystem Function>> | senses user input from HMI |  |
|  | 1987858980.jpg [Provide Climate system with User Input](#_50bd3f022200fcabfc8688251ad91e56) <<System Function>> | Senses user input and gives said input/selection to the Climate system. |  |
|  | 2116425921.jpg [Detect Register Feedback Positions](#_67a2ca2d2a3bd78b04d636a96d8906a1) <<Subsystem Function>> | Senses register feedback positions |  |
|  | -1572298610.jpg [Display Visual Feedback](#_3dc91daae7cbd4657779dc7b5ee3d8fe) <<System Function>> | Shows the user the HMI that visually depicts the register positions and various selections. |  |
|  | 2116425921.jpg [Provide Display Feedback Positions](#_ed274b413a31bf7c929283bdc854c814) <<Subsystem Function>> | detects register positions feedback from register move command |  |

Table 3: List of Logical Functions

## Signal List

Refer to the [Data Dictionary](#_Data_Dictionary) - [Logical Signals](#_Logical_Signals).

# Function Specifications

## Logical Function -775985605.jpg Detect User Input

### Function Overview

#### Function Description

Function is allocated to:

* 44569178.jpg EM Registers <<Logical>>
* 1735144477.jpg Register Controller <<Logical>>

senses user input from HMI

#### Function Variants

*Not supported by MagicDraw report generation.*

|  |  |  |
| --- | --- | --- |
| Variant Name | Variant Description | Variant Condition (optional) |
| e.g. “High Content” |  | e.g. ExtLightTechnology = LED  OR  ExtLightTechnology = Xenon |

#### Input Requirements/Documents

|  |  |  |  |
| --- | --- | --- | --- |
| **Reference**  (Reference as listed in ch. ”References”) | **Section/Requirement** | **Description** | **Derived Requirement**  (optional – reference to requirement in ch. “Function Requirements”) |
| **Feature Requirements** | | | |
|  | <Example:  id + title of relevant Feature Docs> | <Example: “Requirements of Feature …”> | <Note: If you reference a requirement in this column, that requirement should have a trace link in its [“Source”/”Source Req.” attribute](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) field pointing back to the input requirement (or to a requirement inside the input document) given in this table row> |
|  |  |  |  |
| **Ford Engineering Standards** | | | |
|  | <Example: some SDS (requirement)> |  |  |
|  |  |  |  |
| **Legal Regulations** | | | |
|  | Compliance with FMVSS101 | The Feature shall comply with FMVSS101. |  |
|  |  |  |  |
| **Industry Standards** | | | |
|  | ISO 26262 | The system shall be developed according to Ford's implementation of Functional Safety. |  |
|  |  |  |  |
| **Other Sources** | | | |
|  | Air Register Life Cycle from IP-0114 | Registers shall withstand at least 7000 cycles. |  |
|  | Air Register Applied Loads from IP-0117 | Register shall withstand loads of 100N and remain functional. |  |
|  |  |  |  |
|  | Personalizable | EM Registers shall be personalizable. (PPP and/or Preset) |  |
|  | HMI Feedback | EM Registers shall give visual feedback to the User through the HMI. |  |
|  | Registers remember users last chosen position Memory | EM Registers shall remember Users last chosen registers directions. |  |
|  | HVAC Vent Air Flow Direction Interface | EM Registers shall control the direction of air exiting the Registers by interfacing with the HMI. |  |
|  | Fast and Accurate | EM Registers shall provide fast and accurate control of the air vents. |  |
|  | Electrically Actuated | EM Registers has electrically actuated and not human actuated air vents. |  |
|  |  |  |  |

Table 7: Input Requirements/Documents

#### Assumptions

No assumptions specified for this function.

### Function Scope

The -775985605.jpg **– “Detect User Input”** function is called by the following functions:

* 1987858980.jpg – “[Provide Climate system with User Input](#_50bd3f022200fcabfc8688251ad91e56)”

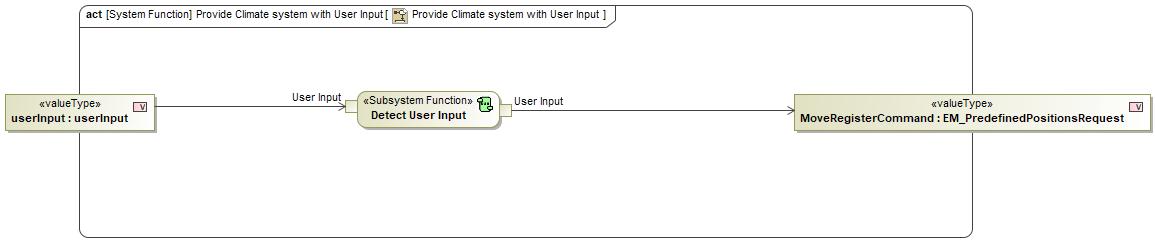


Figure 2: Activity Diagram of 1987858980.jpg “Provide Climate system with User Input” calling -775985605.jpg “Detect User Input”

### Function Interfaces

#### Logical Inputs

|  |  |
| --- | --- |
| **Signal Name** | **Description** |
| User Input  Type:  -1331769109.jpg [userInput](#_92b25d8939283cda19386fd8ddedf11e) | Signal Description:  User input from HMI screen display |

#### Logical Outputs

|  |  |
| --- | --- |
| **Signal Name** | **Description** |
| User Input  Type:  853418517.jpg [EM\_PredefinedPositionsRequest](#_b8deb5335a9beacee07a3dfacdd9df9f) | Signal Description:  Move register commands based on register postion settings or button options |

#### Logical Parameters

*Not supported by MagicDraw report generation.*

|  |  |
| --- | --- |
| **Parameter Name** | **Description** |
| <(Mandatory) Word reference to the “Logical Parameters” name bookmark in the Data Dictionary> | <(Optional) Word reference to the “Logical Parameters” description bookmark in the Data Dictionary> |
|  |  |

### Function Modeling

#### Use Cases

*Not supported by MagicDraw report generation.*

#### State Charts

No state chart associated to specified function.

#### Activity Diagrams

No activity diagram associated to specified function.

#### Sequence Diagrams

No sequence diagram associated to specified function.

#### Decision Tables

### Function Requirements

#### Functional Requirements

##### Normal Operation

Detect User Input

When subsystem function receives the signal "userinput", it shall forward the signal to the subsystem function "Provide User Input"

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | | |
| **Rationale** | Confirm manual, dynamic, or preset register move requests | | | | | | |
| **Acceptance Criteria** | Observation of feedback based on user register move requests | | | | | | |
| **Notes** |  | | | | | | |
| **Source** |  | | | | | **Owner** |  |
| **Source Req.** | * -327200966.jpg HMI System Functionality * -327200966.jpg Display Visual Feedback * -327200966.jpg Body Control System Functionality | | | | | **V&V Method** |  |
| **Type** |  | | | **Priority** | 2 - Medium | **Status** |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | | |

##### Error Handling

No Error Handling Requirements specified.

#### Non-Functional Requirements

No Non-Functional Requirements specified.

#### Functional Safety Requirements

No Functional Safety Requirements specified.

#### Other Requirements

##### Design Requirements

No Design Requirements specified.

#### Uncategorized Requirements

SYNC Range of Coverage Coordinate Box

SYNC range of coverage coordinate box provides the area of coverage for each register relative to the Cockpit as depicted by the Wireframe image

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | | |
| **Rationale** |  | | | | | | |
| **Acceptance Criteria** |  | | | | | | |
| **Notes** |  | | | | | | |
| **Source** |  | | | | | **Owner** |  |
| **Source Req.** |  | | | | | **V&V Method** |  |
| **Type** |  | | | **Priority** |  | **Status** |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | | |

SYNC Coordinate Tacking of Drag Dots

SYNC shall provide coordinate tracking of the Drag Dots between the min and max positions in (X,Y) format, where X is the cross car component of the Drag Dot and Y is the up down component of the Drag Dot

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | | |
| **Rationale** |  | | | | | | |
| **Acceptance Criteria** |  | | | | | | |
| **Notes** |  | | | | | | |
| **Source** |  | | | | | **Owner** |  |
| **Source Req.** |  | | | | | **V&V Method** |  |
| **Type** |  | | | **Priority** |  | **Status** |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | | |

## Logical Function 1987858980.jpg Provide Climate system with User Input

### Function Overview

#### Function Description

Function is allocated to:

* -491569881.jpg Climate System <<Logical>>
* 44569178.jpg EM Registers <<Logical>>
* 44569178.jpg HMI System <<Logical>>

Senses user input and gives said input/selection to the Climate system.

#### Function Variants

*Not supported by MagicDraw report generation.*

|  |  |  |
| --- | --- | --- |
| Variant Name | Variant Description | Variant Condition (optional) |
| e.g. “High Content” |  | e.g. ExtLightTechnology = LED  OR  ExtLightTechnology = Xenon |

#### Input Requirements/Documents

|  |  |  |  |
| --- | --- | --- | --- |
| **Reference**  (Reference as listed in ch. ”References”) | **Section/Requirement** | **Description** | **Derived Requirement**  (optional – reference to requirement in ch. “Function Requirements”) |
| **Feature Requirements** | | | |
|  | <Example:  id + title of relevant Feature Docs> | <Example: “Requirements of Feature …”> | <Note: If you reference a requirement in this column, that requirement should have a trace link in its [“Source”/”Source Req.” attribute](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) field pointing back to the input requirement (or to a requirement inside the input document) given in this table row> |
|  |  |  |  |
| **Ford Engineering Standards** | | | |
|  | <Example: some SDS (requirement)> |  |  |
|  |  |  |  |
| **Legal Regulations** | | | |
|  | Compliance with FMVSS101 | The Feature shall comply with FMVSS101. |  |
|  |  |  |  |
| **Industry Standards** | | | |
|  | ISO 26262 | The system shall be developed according to Ford's implementation of Functional Safety. |  |
|  |  |  |  |
| **Other Sources** | | | |
|  | Air Register Life Cycle from IP-0114 | Registers shall withstand at least 7000 cycles. |  |
|  | Air Register Applied Loads from IP-0117 | Register shall withstand loads of 100N and remain functional. |  |
|  |  |  |  |
|  | Personalizable | EM Registers shall be personalizable. (PPP and/or Preset) |  |
|  | HMI Feedback | EM Registers shall give visual feedback to the User through the HMI. |  |
|  | Registers remember users last chosen position Memory | EM Registers shall remember Users last chosen registers directions. |  |
|  | HVAC Vent Air Flow Direction Interface | EM Registers shall control the direction of air exiting the Registers by interfacing with the HMI. |  |
|  | Fast and Accurate | EM Registers shall provide fast and accurate control of the air vents. |  |
|  | Electrically Actuated | EM Registers has electrically actuated and not human actuated air vents. |  |
|  |  |  |  |

Table 7: Input Requirements/Documents

#### Assumptions

No assumptions specified for this function.

### Function Scope

The 1987858980.jpg **– “Provide Climate system with User Input”** function is called by the following functions:

* 1891945193.jpg – “[Move Registers](#_e69e999f49d30a742ebb3f4b59cf6da8)”
* 347853625.jpg – “[Save LastPosition in Preset](#_58899de816c8ff1f7178a8bba4247812)”
* 1891945193.jpg – “[Store Registers' Positions](#_7986aff96e6ee88b9962283bdc931348)”

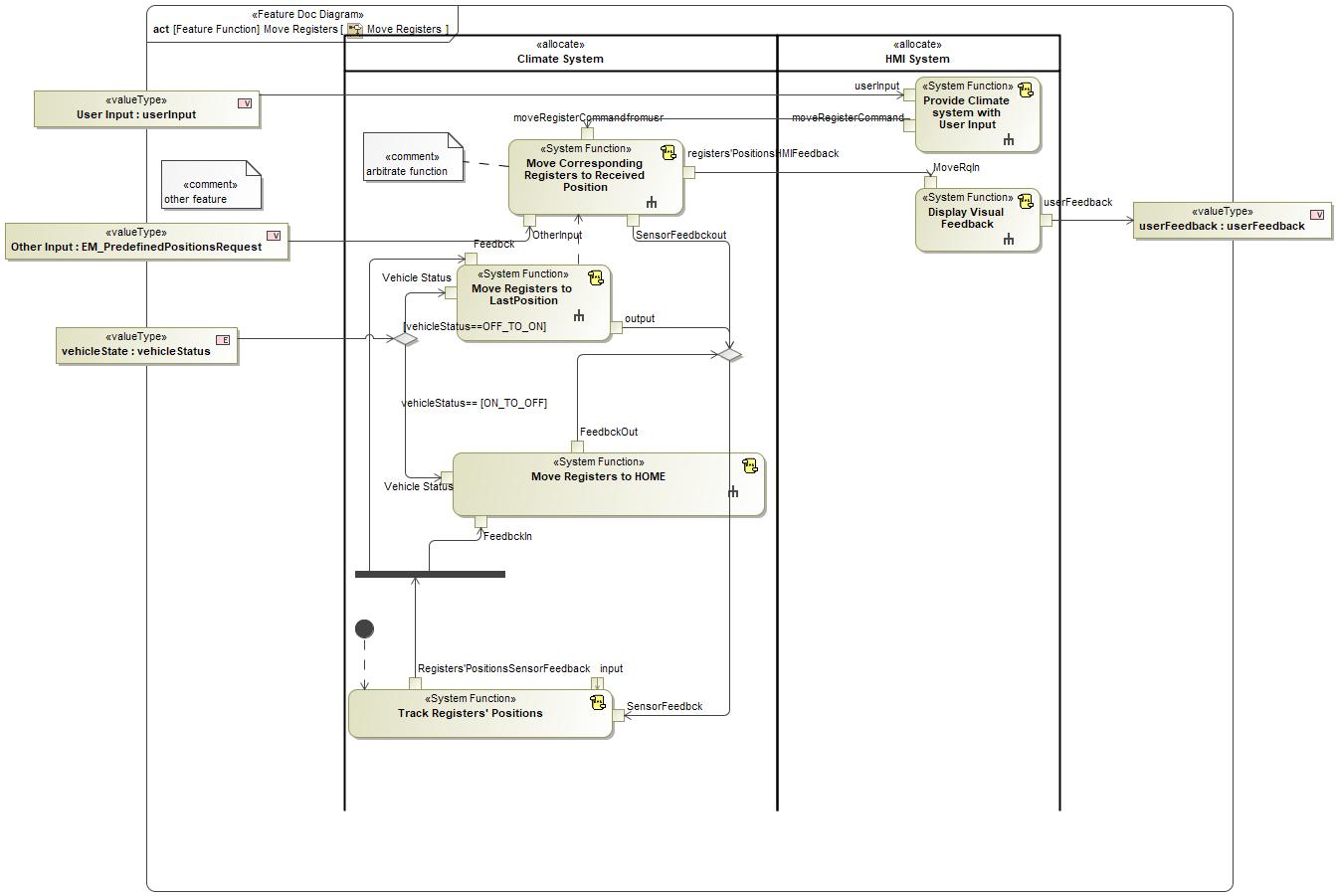


Figure 2: Activity Diagram of 1891945193.jpg “Move Registers” calling 1987858980.jpg “Provide Climate system with User Input”

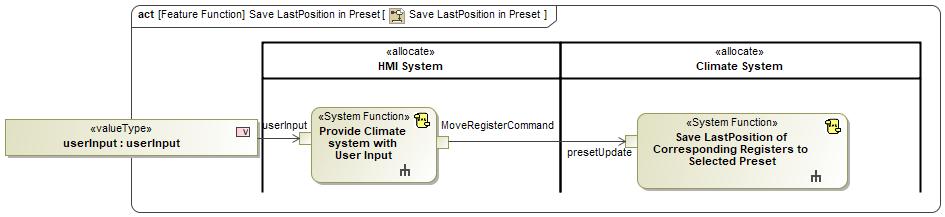


Figure 2: Activity Diagram of 347853625.jpg “Save LastPosition in Preset” calling 1987858980.jpg “Provide Climate system with User Input”

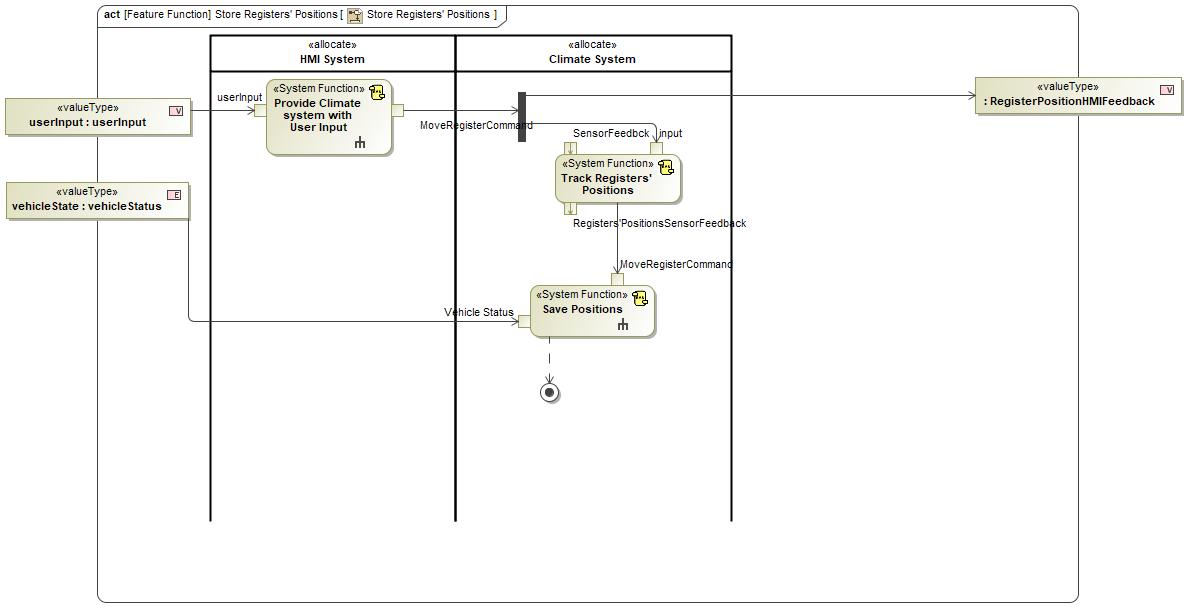


Figure 2: Activity Diagram of 1891945193.jpg “Store Registers' Positions” calling 1987858980.jpg “Provide Climate system with User Input”

### Function Interfaces

#### Logical Inputs

|  |  |
| --- | --- |
| **Signal Name** | **Description** |
| userInput  Type:  -1331769109.jpg [userInput](#_92b25d8939283cda19386fd8ddedf11e) | Signal Description:  User input from HMI screen display |

#### Logical Outputs

|  |  |
| --- | --- |
| **Signal Name** | **Description** |
| MoveRegisterCommand  Type:  853418517.jpg [EM\_PredefinedPositionsRequest](#_b8deb5335a9beacee07a3dfacdd9df9f) | Signal Description:  Move register commands based on register postion settings or button options  Sent to:   * 1987858980.jpg [Track Registers' Positions](#_a2fbd345f943d530c36ffb78beb4b4c2) * -28443014.jpg [Save LastPosition of Corresponding Registers to Selected Preset](#_12d433653b6c1f42157259f6dcd80666) * 363201480.jpg [Move Corresponding Registers to Received Position](#_5219b2dc880b60ac1d10c91499618037) |

#### Logical Parameters

*Not supported by MagicDraw report generation.*

|  |  |
| --- | --- |
| **Parameter Name** | **Description** |
| <(Mandatory) Word reference to the “Logical Parameters” name bookmark in the Data Dictionary> | <(Optional) Word reference to the “Logical Parameters” description bookmark in the Data Dictionary> |
|  |  |

### Function Modeling

#### Use Cases

*Not supported by MagicDraw report generation.*

#### State Charts

No state chart associated to specified function.

#### Activity Diagrams

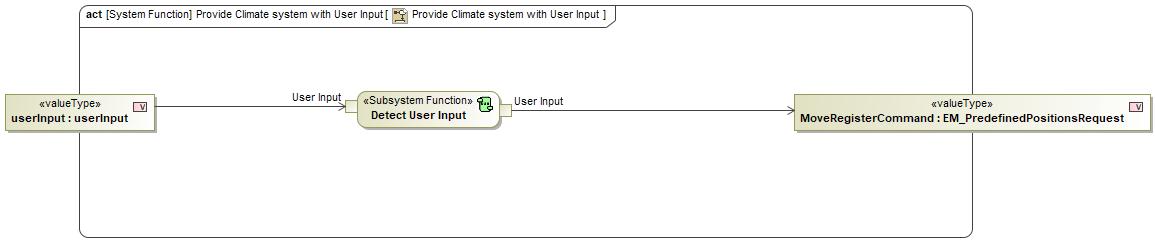


Figure 5: Provide Climate system with User Input

#### Sequence Diagrams

No sequence diagram associated to specified function.

#### Decision Tables

### Function Requirements

#### Functional Requirements

##### Normal Operation

HMI System Functionality

When "Display Visual Feedback" system function receives "RegistersPositionsHMIFeedback, it shall provide "userFeedback".

Provide Climate System with User Input delete function?????

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | | |
| **Rationale** | Provide user move register selections feedback | | | | | | |
| **Acceptance Criteria** | Observation of feedback on HMI Screen | | | | | | |
| **Notes** |  | | | | | | |
| **Source** |  | | | | | **Owner** | George Smith |
| **Source Req.** | * 482733978.jpg EM Reg HMI Functionality * 1405355873.jpg HMI Feedback * 482733978.jpg EM Reg Move Command * 482733978.jpg EM Reg Preset Aim Memo * 1405355873.jpg HVAC Vent Air Flow Direction Interface * 482733978.jpg EM Reg HMI Feedback * 482733978.jpg EM Reg Preset Command | | | | | **V&V Method** |  |
| **Type** |  | | | **Priority** | 2 - Medium | **Status** |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | | |

HMI to Climate System User Input

When "Provide Climate System With User Input" system function receives "MoveRegisterCommand" user input , it shall provide the climate system with "MoveRegisterCommand"

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | | |
| **Rationale** | Process user register move requests | | | | | | |
| **Acceptance Criteria** | Observation of feedback on HMI Screen | | | | | | |
| **Notes** |  | | | | | | |
| **Source** |  | | | | | **Owner** | George Smith |
| **Source Req.** | * 482733978.jpg EM Reg HMI Functionality * 1405355873.jpg Personalizable * 482733978.jpg EM Reg Preset Aim Memo * 1405355873.jpg HVAC Vent Air Flow Direction Interface | | | | | **V&V Method** |  |
| **Type** |  | | | **Priority** | 2 - Medium | **Status** |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | | |

##### Error Handling

No Error Handling Requirements specified.

#### Non-Functional Requirements

No Non-Functional Requirements specified.

#### Functional Safety Requirements

No Functional Safety Requirements specified.

#### Other Requirements

##### Design Requirements

No Design Requirements specified.

## Logical Function 2116425921.jpg Detect Register Feedback Positions

### Function Overview

#### Function Description

Function is allocated to:

* 44569178.jpg EM Registers <<Logical>>
* 1735144477.jpg Register Controller <<Logical>>

Senses register feedback positions

#### Function Variants

*Not supported by MagicDraw report generation.*

|  |  |  |
| --- | --- | --- |
| Variant Name | Variant Description | Variant Condition (optional) |
| e.g. “High Content” |  | e.g. ExtLightTechnology = LED  OR  ExtLightTechnology = Xenon |

#### Input Requirements/Documents

|  |  |  |  |
| --- | --- | --- | --- |
| **Reference**  (Reference as listed in ch. ”References”) | **Section/Requirement** | **Description** | **Derived Requirement**  (optional – reference to requirement in ch. “Function Requirements”) |
| **Feature Requirements** | | | |
|  | <Example:  id + title of relevant Feature Docs> | <Example: “Requirements of Feature …”> | <Note: If you reference a requirement in this column, that requirement should have a trace link in its [“Source”/”Source Req.” attribute](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) field pointing back to the input requirement (or to a requirement inside the input document) given in this table row> |
|  |  |  |  |
| **Ford Engineering Standards** | | | |
|  | <Example: some SDS (requirement)> |  |  |
|  |  |  |  |
| **Legal Regulations** | | | |
|  | Compliance with FMVSS101 | The Feature shall comply with FMVSS101. |  |
|  |  |  |  |
| **Industry Standards** | | | |
|  | ISO 26262 | The system shall be developed according to Ford's implementation of Functional Safety. |  |
|  |  |  |  |
| **Other Sources** | | | |
|  | Air Register Life Cycle from IP-0114 | Registers shall withstand at least 7000 cycles. |  |
|  | Air Register Applied Loads from IP-0117 | Register shall withstand loads of 100N and remain functional. |  |
|  |  |  |  |
|  | Personalizable | EM Registers shall be personalizable. (PPP and/or Preset) |  |
|  | HMI Feedback | EM Registers shall give visual feedback to the User through the HMI. |  |
|  | Registers remember users last chosen position Memory | EM Registers shall remember Users last chosen registers directions. |  |
|  | HVAC Vent Air Flow Direction Interface | EM Registers shall control the direction of air exiting the Registers by interfacing with the HMI. |  |
|  | Fast and Accurate | EM Registers shall provide fast and accurate control of the air vents. |  |
|  | Electrically Actuated | EM Registers has electrically actuated and not human actuated air vents. |  |
|  |  |  |  |

Table 7: Input Requirements/Documents

#### Assumptions

No assumptions specified for this function.

### Function Scope

The 2116425921.jpg **– “Detect Register Feedback Positions”** function is called by the following functions:

* -1572298610.jpg – “[Display Visual Feedback](#_3dc91daae7cbd4657779dc7b5ee3d8fe)”

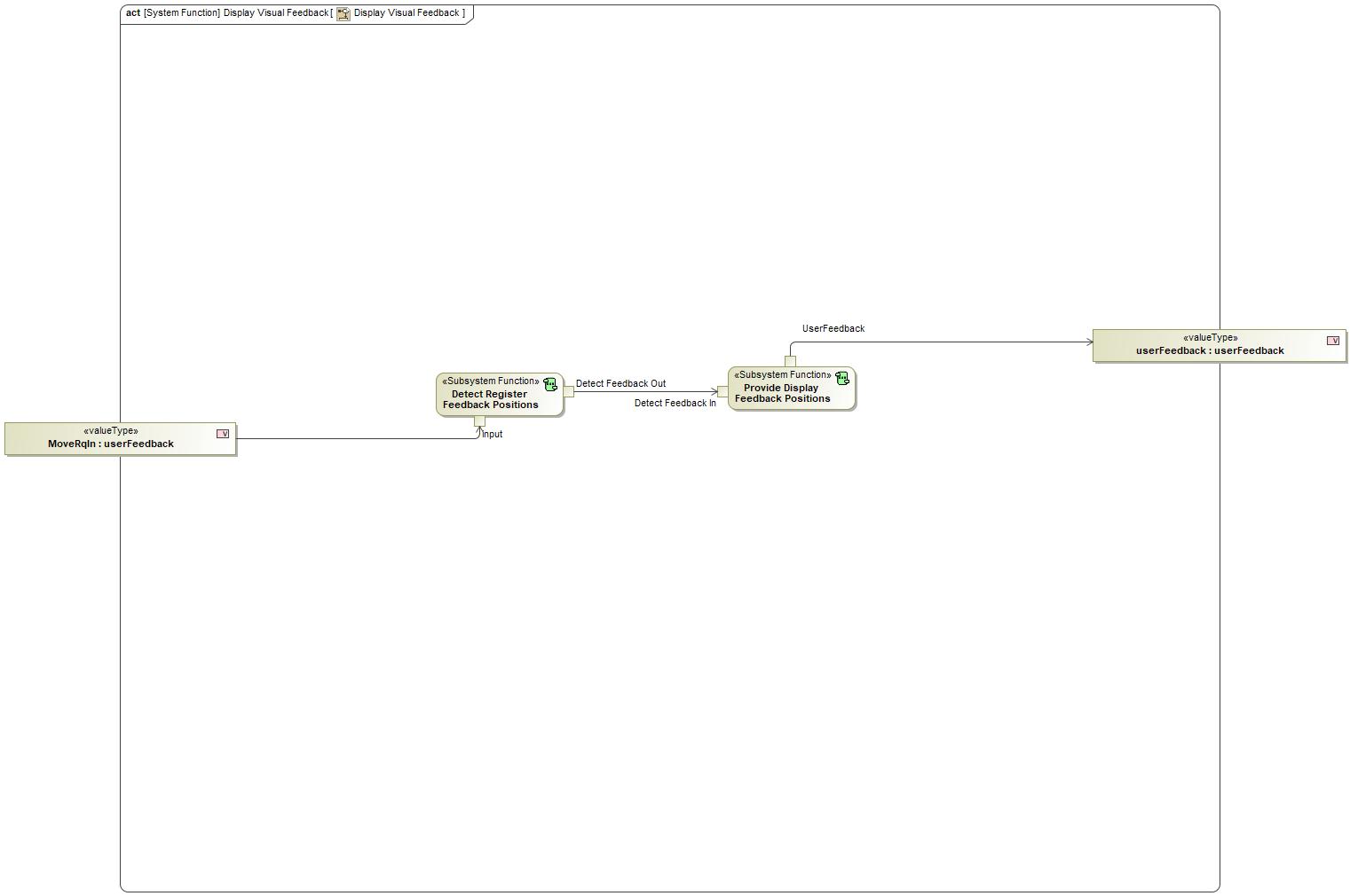


Figure 2: Activity Diagram of -1572298610.jpg “Display Visual Feedback” calling 2116425921.jpg “Detect Register Feedback Positions”

### Function Interfaces

#### Logical Inputs

|  |  |
| --- | --- |
| **Signal Name** | **Description** |
| input  Type:  853418517.jpg [RegisterPositionHMIFeedback](#_a986d850c094bae00c8310d3e8b8c34d) | Signal Description:  Register sensor position feedback to HMI |

#### Logical Outputs

|  |  |
| --- | --- |
| **Signal Name** | **Description** |
| Detect Feedback Out  Type:  853418517.jpg [RegisterPositionHMIFeedback](#_a986d850c094bae00c8310d3e8b8c34d) | Signal Description:  Register sensor position feedback to HMI  Sent to:   * 2116425921.jpg [Provide Display Feedback Positions](#_ed274b413a31bf7c929283bdc854c814) |

#### Logical Parameters

*Not supported by MagicDraw report generation.*

|  |  |
| --- | --- |
| **Parameter Name** | **Description** |
| <(Mandatory) Word reference to the “Logical Parameters” name bookmark in the Data Dictionary> | <(Optional) Word reference to the “Logical Parameters” description bookmark in the Data Dictionary> |
|  |  |

### Function Modeling

#### Use Cases

*Not supported by MagicDraw report generation.*

#### State Charts

No state chart associated to specified function.

#### Activity Diagrams

No activity diagram associated to specified function.

#### Sequence Diagrams

No sequence diagram associated to specified function.

#### Decision Tables

### Function Requirements

#### Functional Requirements

##### Normal Operation

Detect Register Feedback Positions

When system subfunction "Detect Register Feedback Positions" receives the signal "Detect Feedback" it shall forward the signal to the "Provide Display Feedback Positons"

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | | |
| **Rationale** | Provide user updates on register move requests | | | | | | |
| **Acceptance Criteria** | Observation of feedback positions on HMI display screen | | | | | | |
| **Notes** |  | | | | | | |
| **Source** |  | | | | | **Owner** |  |
| **Source Req.** | * -327200966.jpg HMI System Functionality * -327200966.jpg Display Visual Feedback | | | | | **V&V Method** |  |
| **Type** |  | | | **Priority** | 2 - Medium | **Status** |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | | |

##### Error Handling

No Error Handling Requirements specified.

#### Non-Functional Requirements

No Non-Functional Requirements specified.

#### Functional Safety Requirements

No Functional Safety Requirements specified.

#### Other Requirements

##### Design Requirements

No Design Requirements specified.

## Logical Function -1572298610.jpg Display Visual Feedback

### Function Overview

#### Function Description

Function is allocated to:

* 44569178.jpg EM Registers <<Logical>>
* 44569178.jpg HMI System <<Logical>>

Shows the user the HMI that visually depicts the register positions and various selections.

#### Function Variants

*Not supported by MagicDraw report generation.*

|  |  |  |
| --- | --- | --- |
| Variant Name | Variant Description | Variant Condition (optional) |
| e.g. “High Content” |  | e.g. ExtLightTechnology = LED  OR  ExtLightTechnology = Xenon |

#### Input Requirements/Documents

|  |  |  |  |
| --- | --- | --- | --- |
| **Reference**  (Reference as listed in ch. ”References”) | **Section/Requirement** | **Description** | **Derived Requirement**  (optional – reference to requirement in ch. “Function Requirements”) |
| **Feature Requirements** | | | |
|  | <Example:  id + title of relevant Feature Docs> | <Example: “Requirements of Feature …”> | <Note: If you reference a requirement in this column, that requirement should have a trace link in its [“Source”/”Source Req.” attribute](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) field pointing back to the input requirement (or to a requirement inside the input document) given in this table row> |
|  |  |  |  |
| **Ford Engineering Standards** | | | |
|  | <Example: some SDS (requirement)> |  |  |
|  |  |  |  |
| **Legal Regulations** | | | |
|  | Compliance with FMVSS101 | The Feature shall comply with FMVSS101. |  |
|  |  |  |  |
| **Industry Standards** | | | |
|  | ISO 26262 | The system shall be developed according to Ford's implementation of Functional Safety. |  |
|  |  |  |  |
| **Other Sources** | | | |
|  | Air Register Life Cycle from IP-0114 | Registers shall withstand at least 7000 cycles. |  |
|  | Air Register Applied Loads from IP-0117 | Register shall withstand loads of 100N and remain functional. |  |
|  |  |  |  |
|  | Personalizable | EM Registers shall be personalizable. (PPP and/or Preset) |  |
|  | HMI Feedback | EM Registers shall give visual feedback to the User through the HMI. |  |
|  | Registers remember users last chosen position Memory | EM Registers shall remember Users last chosen registers directions. |  |
|  | HVAC Vent Air Flow Direction Interface | EM Registers shall control the direction of air exiting the Registers by interfacing with the HMI. |  |
|  | Fast and Accurate | EM Registers shall provide fast and accurate control of the air vents. |  |
|  | Electrically Actuated | EM Registers has electrically actuated and not human actuated air vents. |  |
|  |  |  |  |

Table 7: Input Requirements/Documents

#### Assumptions

No assumptions specified for this function.

### Function Scope

The -1572298610.jpg **– “Display Visual Feedback”** function is called by the following functions:

* 363201480.jpg – “[Display Visual Feedback](#_22753523c32c985c57b0aefb03cd4528)”
* 1891945193.jpg – “[Move Registers](#_e69e999f49d30a742ebb3f4b59cf6da8)”
* 1891945193.jpg – “[Move Registers to Desired Position](#_3d1006bcc0aa6848ae179b43405435df)”

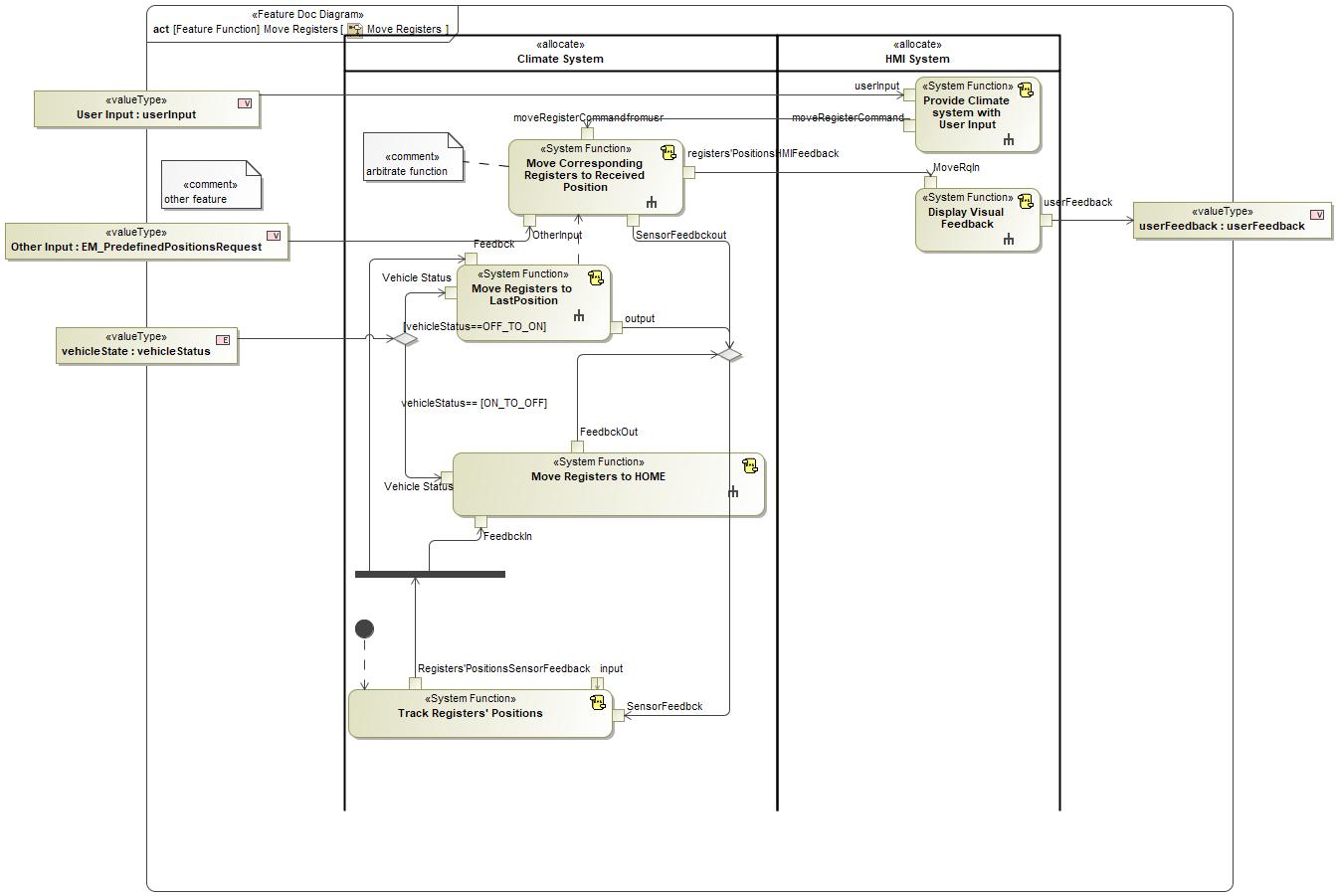


Figure 2: Activity Diagram of 1891945193.jpg “Move Registers” calling -1572298610.jpg “Display Visual Feedback”

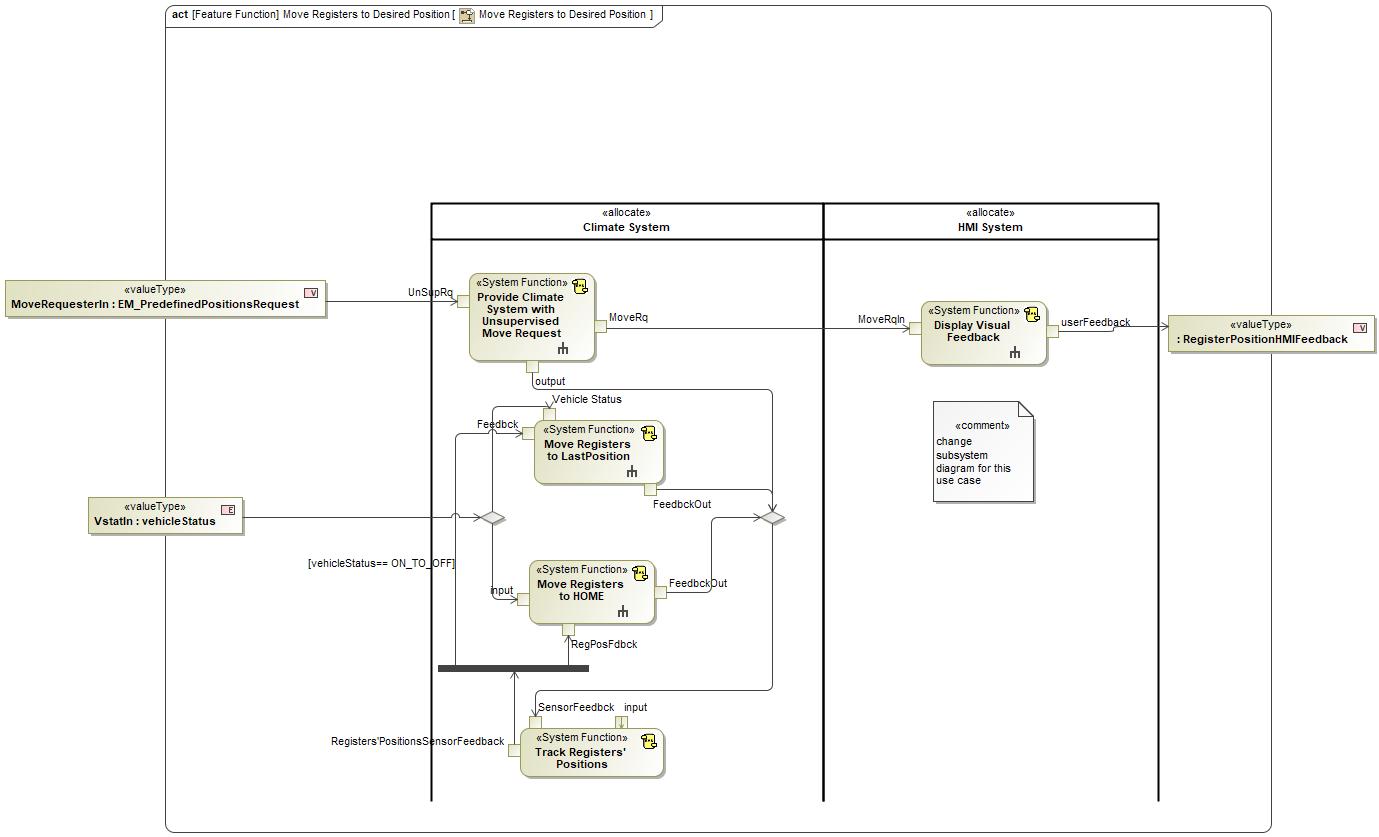


Figure 2: Activity Diagram of 1891945193.jpg “Move Registers to Desired Position” calling -1572298610.jpg “Display Visual Feedback”

### Function Interfaces

#### Logical Inputs

|  |  |
| --- | --- |
| **Signal Name** | **Description** |
| MoveRqIn  Type:  -1331769109.jpg [userFeedback](#_2a0842f9a4e86ee295462fec4e7ecc47) | Signal Description:  User feedback sent to the HMI visual output  Received from:   * 363201480.jpg [Provide Climate System with Unsupervised Move Request](#_b13fd4d8f3af1e75a44c979f520b5f1c) * 363201480.jpg [Move Corresponding Registers to Received Position](#_5219b2dc880b60ac1d10c91499618037) |

#### Logical Outputs

|  |  |
| --- | --- |
| **Signal Name** | **Description** |
| userFeedback  Type:  -1331769109.jpg [userFeedback](#_2a0842f9a4e86ee295462fec4e7ecc47) | Signal Description:  User feedback sent to the HMI visual output |

#### Logical Parameters

*Not supported by MagicDraw report generation.*

|  |  |
| --- | --- |
| **Parameter Name** | **Description** |
| <(Mandatory) Word reference to the “Logical Parameters” name bookmark in the Data Dictionary> | <(Optional) Word reference to the “Logical Parameters” description bookmark in the Data Dictionary> |
|  |  |

### Function Modeling

#### Use Cases

*Not supported by MagicDraw report generation.*

#### State Charts

No state chart associated to specified function.

#### Activity Diagrams

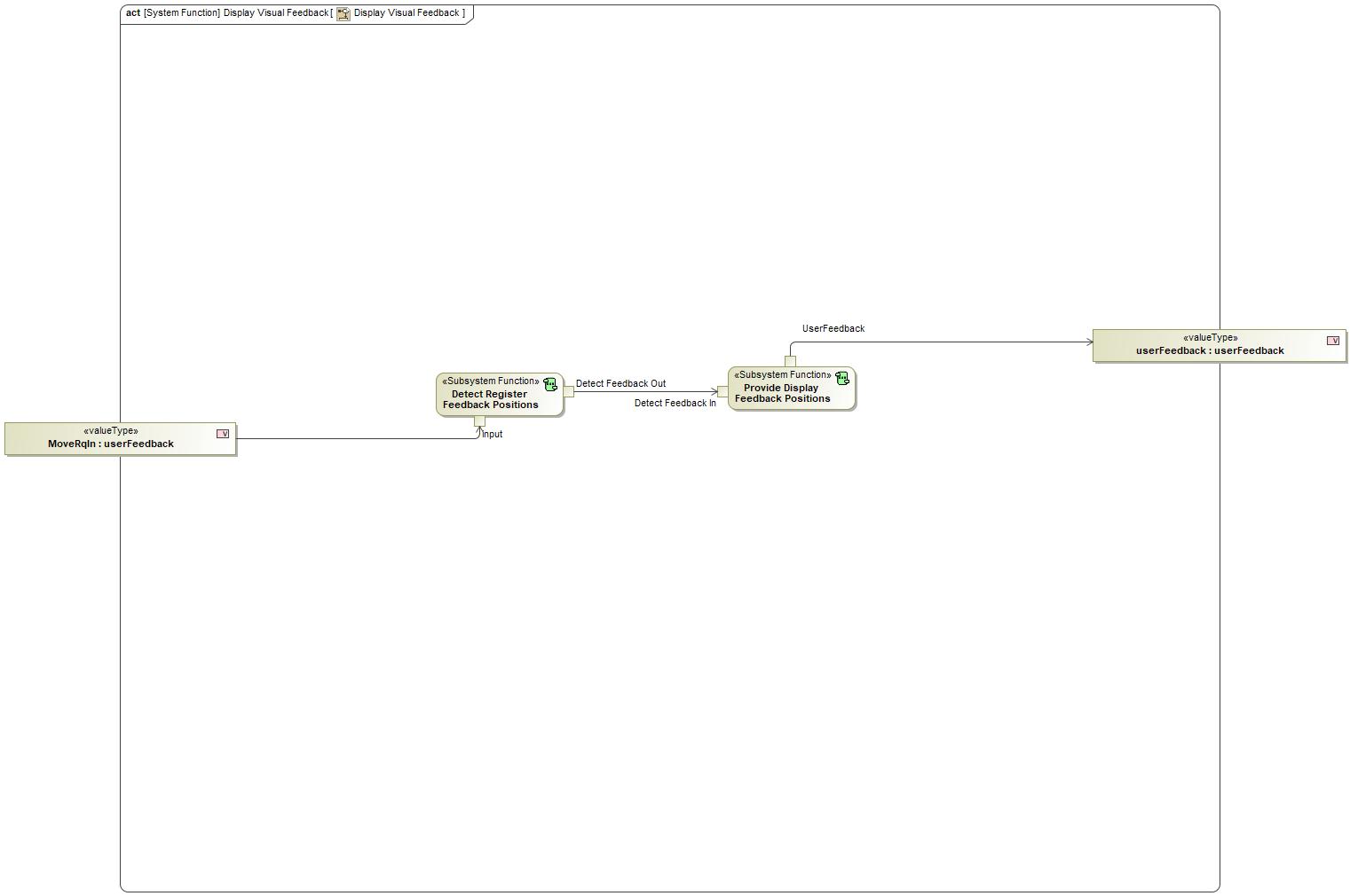


Figure 5: Display Visual Feedback

#### Sequence Diagrams

No sequence diagram associated to specified function.

#### Decision Tables

### Function Requirements

#### Functional Requirements

##### Normal Operation

HMI System Functionality

When "Display Visual Feedback" system function receives "RegistersPositionsHMIFeedback, it shall provide "userFeedback".

Provide Climate System with User Input delete function?????

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | | |
| **Rationale** | Provide user move register selections feedback | | | | | | |
| **Acceptance Criteria** | Observation of feedback on HMI Screen | | | | | | |
| **Notes** |  | | | | | | |
| **Source** |  | | | | | **Owner** | George Smith |
| **Source Req.** | * 482733978.jpg EM Reg HMI Functionality * 1405355873.jpg HMI Feedback * 482733978.jpg EM Reg Move Command * 482733978.jpg EM Reg Preset Aim Memo * 1405355873.jpg HVAC Vent Air Flow Direction Interface * 482733978.jpg EM Reg HMI Feedback * 482733978.jpg EM Reg Preset Command | | | | | **V&V Method** |  |
| **Type** |  | | | **Priority** | 2 - Medium | **Status** |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | | |

Display Visual Feedback

When "Display Visual Feedback" system function receives the input "RegistersPositionsHMIFeedback, it shall display the status of EM registers to the user through HMI.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | | |
| **Rationale** | Provide user with acknowledgement of requested register move commands | | | | | | |
| **Acceptance Criteria** | Observation of feedback to HMI Screen | | | | | | |
| **Notes** |  | | | | | | |
| **Source** |  | | | | | **Owner** | George Smith |
| **Source Req.** | * 1405355873.jpg HMI Feedback * 482733978.jpg EM Reg HMI Feedback | | | | | **V&V Method** |  |
| **Type** |  | | | **Priority** | 2 - Medium | **Status** |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | | |

##### Error Handling

No Error Handling Requirements specified.

#### Non-Functional Requirements

No Non-Functional Requirements specified.

#### Functional Safety Requirements

No Functional Safety Requirements specified.

#### Other Requirements

##### Design Requirements

No Design Requirements specified.

## Logical Function 2116425921.jpg Provide Display Feedback Positions

### Function Overview

#### Function Description

Function is allocated to:

* 44569178.jpg EM Registers <<Logical>>
* 1735144477.jpg Register Controller <<Logical>>

detects register positions feedback from register move command

#### Function Variants

*Not supported by MagicDraw report generation.*

|  |  |  |
| --- | --- | --- |
| Variant Name | Variant Description | Variant Condition (optional) |
| e.g. “High Content” |  | e.g. ExtLightTechnology = LED  OR  ExtLightTechnology = Xenon |

#### Input Requirements/Documents

|  |  |  |  |
| --- | --- | --- | --- |
| **Reference**  (Reference as listed in ch. ”References”) | **Section/Requirement** | **Description** | **Derived Requirement**  (optional – reference to requirement in ch. “Function Requirements”) |
| **Feature Requirements** | | | |
|  | <Example:  id + title of relevant Feature Docs> | <Example: “Requirements of Feature …”> | <Note: If you reference a requirement in this column, that requirement should have a trace link in its [“Source”/”Source Req.” attribute](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) field pointing back to the input requirement (or to a requirement inside the input document) given in this table row> |
|  |  |  |  |
| **Ford Engineering Standards** | | | |
|  | <Example: some SDS (requirement)> |  |  |
|  |  |  |  |
| **Legal Regulations** | | | |
|  | Compliance with FMVSS101 | The Feature shall comply with FMVSS101. |  |
|  |  |  |  |
| **Industry Standards** | | | |
|  | ISO 26262 | The system shall be developed according to Ford's implementation of Functional Safety. |  |
|  |  |  |  |
| **Other Sources** | | | |
|  | Air Register Life Cycle from IP-0114 | Registers shall withstand at least 7000 cycles. |  |
|  | Air Register Applied Loads from IP-0117 | Register shall withstand loads of 100N and remain functional. |  |
|  |  |  |  |
|  | Personalizable | EM Registers shall be personalizable. (PPP and/or Preset) |  |
|  | HMI Feedback | EM Registers shall give visual feedback to the User through the HMI. |  |
|  | Registers remember users last chosen position Memory | EM Registers shall remember Users last chosen registers directions. |  |
|  | HVAC Vent Air Flow Direction Interface | EM Registers shall control the direction of air exiting the Registers by interfacing with the HMI. |  |
|  | Fast and Accurate | EM Registers shall provide fast and accurate control of the air vents. |  |
|  | Electrically Actuated | EM Registers has electrically actuated and not human actuated air vents. |  |
|  |  |  |  |

Table 7: Input Requirements/Documents

#### Assumptions

No assumptions specified for this function.

### Function Scope

The 2116425921.jpg **– “Provide Display Feedback Positions”** function is called by the following functions:

* -1572298610.jpg – “[Display Visual Feedback](#_3dc91daae7cbd4657779dc7b5ee3d8fe)”

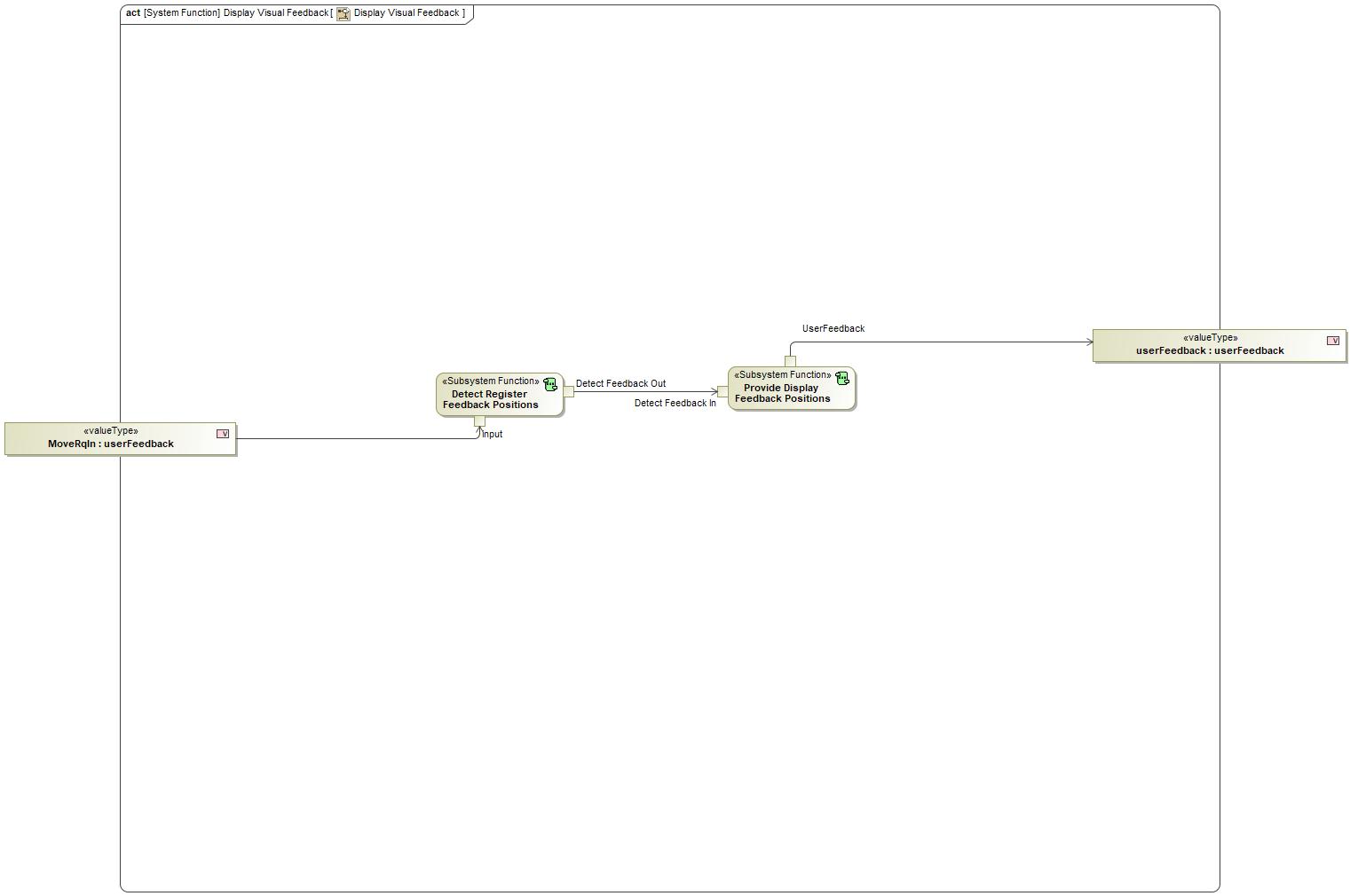


Figure 2: Activity Diagram of -1572298610.jpg “Display Visual Feedback” calling 2116425921.jpg “Provide Display Feedback Positions”

### Function Interfaces

#### Logical Inputs

|  |  |
| --- | --- |
| **Signal Name** | **Description** |
| Detect Feedback In  Type:  853418517.jpg [RegisterPositionHMIFeedback](#_a986d850c094bae00c8310d3e8b8c34d) | Signal Description:  Register sensor position feedback to HMI  Received from:   * 2116425921.jpg [Detect Register Feedback Positions](#_67a2ca2d2a3bd78b04d636a96d8906a1) |

#### Logical Outputs

|  |  |
| --- | --- |
| **Signal Name** | **Description** |
| UserFeedback  Type:  -1331769109.jpg [userFeedback](#_2a0842f9a4e86ee295462fec4e7ecc47) | Signal Description:  User feedback sent to the HMI visual output |

#### Logical Parameters

*Not supported by MagicDraw report generation.*

|  |  |
| --- | --- |
| **Parameter Name** | **Description** |
| <(Mandatory) Word reference to the “Logical Parameters” name bookmark in the Data Dictionary> | <(Optional) Word reference to the “Logical Parameters” description bookmark in the Data Dictionary> |
|  |  |

### Function Modeling

#### Use Cases

*Not supported by MagicDraw report generation.*

#### State Charts

No state chart associated to specified function.

#### Activity Diagrams

No activity diagram associated to specified function.

#### Sequence Diagrams

No sequence diagram associated to specified function.

#### Decision Tables

### Function Requirements

#### Functional Requirements

##### Normal Operation

Provide Display Feedback Positions

When subsystem function receives the signal 'Register Positions Feedback', it shall forward the signal "UserFeedback' to the HMI display

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Requirement ID: | | | | | | | |
| **Rationale** | Provide user register move position feedback | | | | | | |
| **Acceptance Criteria** | Observation of feedback to HMI display screen | | | | | | |
| **Notes** |  | | | | | | |
| **Source** |  | | | | | **Owner** |  |
| **Source Req.** | * -327200966.jpg HMI System Functionality * -327200966.jpg Display Visual Feedback | | | | | **V&V Method** |  |
| **Type** |  | | | **Priority** | 2 - Medium | **Status** |  |
| [Req. Template](http://wiki.ford.com/display/RequirementsEngineering/Requirements+Attributes) Version | | 6.0 | End of Requirement | | | | |

##### Error Handling

No Error Handling Requirements specified.

#### Non-Functional Requirements

No Non-Functional Requirements specified.

#### Functional Safety Requirements

No Functional Safety Requirements specified.

#### Other Requirements

##### Design Requirements

No Design Requirements specified.

# Open Concerns

| ID | Concern Description | e-Tracker / Reference | Responsible | Status | Solution |
| --- | --- | --- | --- | --- | --- |
| 1 |  |  |  |  |  |

Table 8: Open Concerns *(Not supported by MagicDraw report generation.)*

# Revision History

| Rev.  (revision) | Date | Description | Approved by | Responsible |
| --- | --- | --- | --- | --- |
| FS1 | 2021-03-04 | 3/4/2021 added revision level to generated documents |  | mimhof4 |

## Template Revisions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Rev. | Date | Description | Responsible |
| 1 | 0 | 2016-02-26 | Initial version, derived form FDS | Jbaden1 |
| 1 | 1 | 2016-02-26 | Word properties corrected | Jbaden1 |
| 1 | 2 | 2016-03-10 | Clean up of ocument meta data (Word properties) | Jbaden1 |
| 1 | 3 | 2016-03-22 | * Footer formating corrected (Issue 19) * “Constraints” chapter renamed to “Input Requirements” (Issue 20) | Jbaden1 |
| 1 | 4 | 2016-04-20 | * Broken Wiki links repaired | Jbaden1 |
| 2 | 0 | 2016-06-10 | * Document metadata adapted. Prepared for new macros * DTC table removed * HMI function added as a chapter (details still to be refined) * Signal / Parameter IDs column deleted interface tables | Jbaden1 |
| 2 | 1 | 2016-07-14 | * Converted to SysML diagrams * HMI section further elaborated * Template version added to footer * Dedicated Startup / Shutdown sections removed (only hints added) * Data Dictionary reworked and Signal / Parameter IDs column re-introduced | Jbaden1 |
| 2 | 2 | 2016-12-07 | * Minor formatting changes | Jbaden1 |
| 3 |  |  | Skipped to synchronize with Specification\_Macros.dotm |  |
| 4 |  |
| 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 | * Some additional hints. * Hyperlinks highlighted in hints | Jbaden1 |
| 6 | 0 | 2017-04-28 | * Editorial change. Hints added to chapter 4.1.4 * Chapter “Traceability Matrix” removed | Jbaden1 |
| 6 | 0 | 2018-04-28 | * CR69/63: New chapters added for Functional Safety (FTTI and Technical Safety Requirements) * CR53: New coversheet + additional meta-data * CR76: merge sections for configuration and for calibration parameters into one on Function Level | Jbaden1 |
| 6 | 0 | 2018-08-06 | * CR66: Fix version numbering in footer of Function Spec | 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 | 2018-11-12 | * Explanatory text in Variants” section revised * Functional Safety modifications as agreed with FuSa core team (Baseline: November 2018 Dearborn On-Site) | Jbaden1 |
| 6 | 0b | 2020-02-10 | Bugfix release:   * Chapter “Decomposed FSRs” renamed to Functional Safety Requirements”. ASIL decomposition table removed, no longer supported on Function Level. Got also be corrupted, when imported to VSEM * Refinement of FSRs no longer supported by Function Specification (as requested by Functional Safety team). FSR chapter just forwards FSRs from the Feature Docs 1:1 to the Implemented Function(s). | Jbaden1 |
| 6 | 1a | 2019-01-02 | * Editorial changes (in “Variants” section) | Jbaden1 |
| 6 | 1a | 2019-01-21 | * Template Id set to 2 | Jbaden1 |
| 6 | 1a | 2019-03-22 | * Chapter “Decomposed FSRs” renamed to Functional Safety Requirements” A new chapter “ASIL Decomposition of of Functional Safety Requirements” added as a subsection to that chapter. | Jbaden1 |
| 6 | 1a | 2019-04-05 | * Some wording in ASIL decomposition table modified. Description of fields in that table improved. | Jbaden1 |
| 6 | 1a | 2019-04-05 | * ASIL decomposition table removed (ASIL decomposition only allowed on Feature Level or on Component Level in the FIS or ECU Functional Spec) | Jbaden1 |
| 6 | 1a | 2019-07-02 | * "Important" box added on cover sheet which points to the macros * Chapters “References” and “Glossary” moved back up to section “Introduction * Chapter “Inputs Requirements” reworked | Jbaden1 |
| 6 | 1a | 2019-09-10 | * Chapter 4.1.4 has now one section per modeling technique again. This is to allow more intuitive tailoring of the section (driven by AV team request). | Jbaden1 |
| 6 | 1a | 2019-09-10 | * 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 | 2019-05-11 | * Per FuSa core team request: Subsection “Functional Safety Requirements” removed completely from Function Spec. FSRs are only captured in the Feature Doc * Chapter “Input Requirements” renamed to “Input Requirements/Documents” | Jbaden1 |
| 6 | 1a | 2019-12-05 | * Upstream Documents section added to “Input Requirements/Documents” table * Custom style table formatting removed * Per FuSa core team request: Subsection “Functional Safety Requirements” to remain in (but no ASIL Decomposition in Function Spec) | Jbaden1 |
| 6 | 1a | 2019-12-09 | * Term “Upstream Documents” replaced by “Feature Requirements” in “Input Requirements/Documents” table * ASIL Decomposition table replaced by a version, which get not corrupted during VSEM import. | Jbaden1 |
| 6 | 1a | 2019-12-10 | * Refinement of FSRs no longer supported by Function Specification (as requested by Functional Safety team). FSR chapter just forwards FSRs from the Feature Docs 1:1 to the Implemented Function(s). | Jbaden1 |
| 6 | 1a | 2019-12-10 | Minor changes made to enable use of the function group spec for specification of system services (system service spec is to be removed):   * Chapter “Functional Architecture” renamed to “Functional Decomposition and Architecture” * Functional Decomposition diagram added to renamed chapter. | Jbaden1 |
| 6 | 1a | 2020-02-12 | Minor rewording of hint for FSR table after review by FuSa team. | 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 * Copyright date field on cover sheet corrected | Jbaden1 |

# Appendix

## Data Dictionary

### Logical Signals

Actuate Register Command

Signal requesting actuation of register position changes

|  |  |  |
| --- | --- | --- |
| **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 Actuate Register Command

EM\_DesiredPositionsRequest

Manual move register requests

|  |  |  |
| --- | --- | --- |
| **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 EM\_DesiredPositionsRequest

RegisterPositionHMIFeedback

Register sensor position feedback to HMI

|  |  |  |
| --- | --- | --- |
| **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 RegisterPositionHMIFeedback

RegisterPositionsSensorFeedback

Signal to provide register position feedback to actuator

|  |  |  |
| --- | --- | --- |
| **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 RegisterPositionsSensorFeedback

vehicleStatus

Vehicle on off status used by register move commands

|  |  |  |
| --- | --- | --- |
| **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 vehicleStatus

### Logical Parameters

### Encoding Types

SensFdbkLHIBHoriz

Sense left hand horizontal inboard register vane positions and forward to RCCM

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of SensFdbkLHIBHoriz

RegisterPositionsSensorFeedback

Signal to provide register position feedback to actuator

**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) |  |  |
| **SensorFeedbck\_LHOB\_Hor** |  |
| **SensorFeedbck\_LHOB\_Ver** |  |
| **SensorFeedbck\_LHIB\_Hor** |  |
| **SensorFeedbck\_LHIB\_Ver** |  |
| **SensorFeedbck\_RHIB\_Hor** |  |
| **SensorFeedbck\_RHIB\_Ver** |  |
| **SensorFeedbck\_RHOB\_Hor** |  |
| **SensorFeedbck\_RHOB\_Ver** |  |
| **Unit** | |  |

Table: Encoding Details of RegisterPositionsSensorFeedback

Register\_RHIB

Register location passenger inboard

**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) |  |  |
| **NoChange** | Logical signal for register RHIB no change position |
| **ManualChange** | Logical signal for register RHIB manual change position |
| **FeatureChange** | Logical signal for register RHIB Feature change position |
| **Open** | Logical signal for register RHIB open position |
| **Close** | Logical signal for register RHIB close position |
| **Unit** | |  |

Table: Encoding Details of Register\_RHIB

ActiveButtonsLHS

Feedback active button options for the left side of the 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) |  |  |
| **NonePressed** | Logical signal active button LHS for none pressed |
| **OnBodyLHS** | Logical signal active button LHS for on body |
| **OffBodyLHS** | Logical signal active button LHS for off body |
| **PresetOneLHS** | Logical signal active button LHS for preset one |
| **PresetTwoLHS** | Logical signal active button LHS for preset two |
| **PresetOneSaveLHS** | Logical signal active button LHS for preset one save |
| **PresetTwoSaveLHS** | Logical signal active button LHS for preset two save |
| **Cycle\_8\_LHS** | Logical signal active button LHS for cycling figure 8 |
| **Cycle\_C\_LHS** | Logical signal active button LHS for close |
| **Cycle\_O\_LHS** | Logical signal active button LHS for open |
| **Cycle\_--\_LHS** | Logical signal active button LHS for cycling horizontal |
| **Cycle\_|\_LHS** | Logical signal active button LHS for cycling vertical |
| **Unit** | |  |

Table: Encoding Details of ActiveButtonsLHS

Request LIN

Place holder for LIN signals 707 program

**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 LIN

SensorFeedBack\_RHOB\_Vertical

Register sensor position feedback to HMI for right hand outboard register vertical voltage

**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 SensorFeedBack\_RHOB\_Vertical

SensFdbkRHIBHoriz

Sense right hand horizontal inboard register vane positions and forward to RCCM

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of SensFdbkRHIBHoriz

ActCmdLHIBHoriz

Signal from RCCM to left hand inboard horizontal vane register actuator

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of ActCmdLHIBHoriz

ActiveButtonsRHS

Feedback for active button options for the right side of the 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) |  |  |
| **NonePressed** | Logical signal active button RHS for none pressed |
| **OnBodyLHS** | Logical signal active button RHS for on body |
| **OffBodyLHS** | Logical signal active button RHS for off body |
| **PresetOneLHS** | Logical signal active button RHS for preset one |
| **PresetTwoLHS** | Logical signal active button RHS for preset two |
| **PresetOneSaveLHS** | Logical signal active button RHS for preset one save |
| **PresetTwoSaveLHS** | Logical signal active button RHS for preset two save |
| **Cycle\_8\_LHS** | Logical signal active button RHS for cycling figure 8 |
| **Cycle\_C\_LHS** | Logical signal active button RHS for close |
| **Cycle\_O\_LHS** | Logical signal active button RHS for open |
| **Cycle\_--\_LHS** | Logical signal active button RHS for cycling horizontal |
| **Cycle\_|\_LHS** | Logical signal active button RHS for cycling vertical |
| **Unit** | |  |

Table: Encoding Details of ActiveButtonsRHS

Feedback LIN

Place holder for LIN feedback signals 707 program

**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 Feedback LIN

ActCmdRHOBVert

Signal from RCCM to right hand outboard vertical vane register actuator

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of ActCmdRHOBVert

RgstrVertObl\_An\_Actl

Aim Status of Righthand Inboard Register Vertical Vanes

**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) |  |  |
| **unnamed1** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrVertObl\_An\_Actl

Register\_RHOB\_Horizontal

Register airflow direction up or 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) |  |  |
| **Percentage** | Logical signal for register RHOB horizontal percentage change position horizontal |
| **Unit** | |  |

Table: Encoding Details of Register\_RHOB\_Horizontal

RgstrHzntlObl\_An\_Rq

Command Position of Horizontal Vanes for Righthand Outboard Register

**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) |  |  |
| **Deg** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrHzntlObl\_An\_Rq

RgstrHzntllbr\_An\_Actl

Aim Status of Righthand Inboard Register Horizontal Vanes

**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 RgstrHzntllbr\_An\_Actl

SensedRegPosRHOBvert

**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 SensedRegPosRHOBvert

Feedback Sensor to RCCM

**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) |  |  |
| **Voltage** |  |
| **Unit** | |  |

Table: Encoding Details of Feedback Sensor to RCCM

RgstrHzntllbr\_An\_Rq

Command Position of Horizontal Vanes for Righthand Inboard Register

**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 RgstrHzntllbr\_An\_Rq

Sensed\_Actuator\_Position

Signal Not Used

**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 Sensed\_Actuator\_Position

ActCmdLHOBHorizLin

Place holder for LIN signals

**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 ActCmdLHOBHorizLin

OpenCloseRequest

Move request to close or open register vertical vanes

**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) |  |  |
| **NoChange** |  |
| **Open** |  |
| **Close** |  |
| **Unit** | |  |

Table: Encoding Details of OpenCloseRequest

SensFdbkLHOBVert

Sense left hand vertical outboard register vane positions and forward to RCCM

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of SensFdbkLHOBVert

EM\_DesiredPositionsRequest

Manual move register requests

**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) |  |  |
| **register\_LHOB\_Horz** |  |
| **register\_LHOB\_Vert** |  |
| **register\_LHIB\_Vert** |  |
| **register\_LHIB\_Horz** |  |
| **register\_RHOB\_Horz** |  |
| **register\_RHOB\_Vert** |  |
| **register\_RHIB\_Vert** |  |
| **register\_RHIB\_Horz** |  |
| **requesterType** |  |
| **Unit** | |  |

Table: Encoding Details of EM\_DesiredPositionsRequest

ActCmdLHIBHorizLin

Place holder for LIN signals

**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 ActCmdLHIBHorizLin

SensorFeedBack\_LHIB\_Vertical

Register sensor position feedback to HMI for Left hand inboard register voltage

**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 SensorFeedBack\_LHIB\_Vertical

Register\_RHIB\_Horizontal

Register airflow direction up or 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) |  |  |
| **Percentage** | Logical signal for register RHIB horizontal percentage change position horizontal |
| **Unit** | |  |

Table: Encoding Details of Register\_RHIB\_Horizontal

ActCmdLHIBVert

Signal from RCCM to left hand inboard vertica vane register actuator

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of ActCmdLHIBVert

HMIFeedback\_RHIB\_Vertical

Register position feedback to HMI

**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 HMIFeedback\_RHIB\_Vertical

SensorFeedback\_RHIB\_Horizontal

Register sensor position feedback to HMI for right hand inboard register horizontal voltage

**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 SensorFeedback\_RHIB\_Horizontal

SensedRegPosLHIBvert

**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 SensedRegPosLHIBvert

RgstrVertObr\_An\_Rq

Command Position of Vertical Vanes for Righthand Outboard Register

**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) |  |  |
| **unnamed1** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrVertObr\_An\_Rq

SensFdbkLHIBVert

Sense left hand vertical inboard register vane positions and forward to RCCM

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of SensFdbkLHIBVert

Sensed Register Positon

**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) |  |  |
| **Sensed Register Pos** |  |
| **Unit** | |  |

Table: Encoding Details of Sensed Register Positon

SensFdbkRHOBVert

Sense right hand vertical inboard register vane positions and forward to RCCM

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of SensFdbkRHOBVert

SensorFeedback\_LHIB\_Horizontal

Register sensor position feedback to HMI for left hand inboard register voltage

**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 SensorFeedback\_LHIB\_Horizontal

userInput

User input from HMI screen display

**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) |  |  |
| **HMI Touch Input** | Logical signal that provides user input from HMI Touch screen |
| **Unit** | |  |

Table: Encoding Details of userInput

RgstrVertIbl\_An\_Actl

Aim Status of Lefthand Inboard Register Vertical Vanes

**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) |  |  |
| **unnamed1** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrVertIbl\_An\_Actl

Feedback HRD

Place holder for hardwire feedback from potentiometer to actuator

**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 Feedback HRD

RgstrSetObl\_D\_Rq

Change Aim Command for Lefthand Outboard Register

**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) |  |  |
| **0x0** |  |
| **0x1** |  |
| **0x2** |  |
| **0x3** |  |
| **0x4** |  |
| **0x5** |  |
| **0x6** |  |
| **0x7** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrSetObl\_D\_Rq

ActCmdRHIBHoriz

Signal from RCCM to right hand inboard horizontal vane register actuator

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of ActCmdRHIBHoriz

Register\_LHIB\_Horizontal

Register up or down airflow direction

**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) |  |  |
| **Percentage** | Logical signal for register LHIB horizontal percentage change position horizontal |
| **Unit** | |  |

Table: Encoding Details of Register\_LHIB\_Horizontal

SensFdbkRHOBHoriz

Sense right hand horizontal outboard register vane positions and forward to RCCM

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of SensFdbkRHOBHoriz

HMIFeedback\_LHIB\_Vertical

Register position feedback to HMI

**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 HMIFeedback\_LHIB\_Vertical

APIM to RCCM ActvButtn

**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) |  |  |
| **ActvButtnLeft\_D\_Rq** |  |
| **ActButtnRight\_D\_Rq** |  |
| **Unit** | |  |

Table: Encoding Details of APIM to RCCM ActvButtn

HMIFeedback\_RHOB

Register position feedback to HMI

**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) |  |  |
| **Position** | Logical signal HMI feedback RHOB position |
| **Close** | Logical signal HMI feedback RHOB close |
| **Unit** | |  |

Table: Encoding Details of HMIFeedback\_RHOB

SensedRegPosRHIBvert

**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 SensedRegPosRHIBvert

RgstrVertlbl\_An\_Rq

Command Position of Vertical Vanes for Lefthand Inboard Register

**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) |  |  |
| **unnamed1** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrVertlbl\_An\_Rq

SensFdbkRHIBVert

Sense right hand vertical inboard register vane positions and forward to RCCM

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of SensFdbkRHIBVert

ActCmdLHIBVertLin

Place holder for LIN signals

**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 ActCmdLHIBVertLin

Active\_ButtonFeedbck

**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 Buttons LHS** |  |
| **Active Buttons RHS** |  |
| **Unit** | |  |

Table: Encoding Details of Active\_ButtonFeedbck

ActuatorCommand\_LHIB\_Horizontal

Left hand inboard register voltage for up down 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) |  |  |
| **Voltage** | Logical signal for LHIB horizontal percentage change |
| **Unit** | |  |

Table: Encoding Details of ActuatorCommand\_LHIB\_Horizontal

Register\_LHOB\_Horizontal

Register up or down airflow direction

**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) |  |  |
| **Degrees** | Logical signal for register LHOB horizontal percentage change position horizontal |
| **Unit** | |  |

Table: Encoding Details of Register\_LHOB\_Horizontal

RgstrSetIbr\_D\_Rq

Change Aim Command for Righthand Intboard Register

**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) |  |  |
| **0x0** |  |
| **0x1** |  |
| **0x2** |  |
| **0x3** |  |
| **0x4** |  |
| **0x5** |  |
| **0x6** |  |
| **0x7** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrSetIbr\_D\_Rq

ActuatorCommand\_LHOB\_Horizontal

Left hand inboard register voltage for up down 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) |  |  |
| **Voltage** | Logical signal for LHOB horizontal percentage change |
| **Unit** | |  |

Table: Encoding Details of ActuatorCommand\_LHOB\_Horizontal

ActCmdLHOBHoriz

Signal from RCCM to left hand inboard horizontal vane register actuator

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of ActCmdLHOBHoriz

RgstrVertllbr\_An\_Actl

Aim Status of Righthand Inboard Register Vertical Vanes

**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) |  |  |
| **RotationAngle** |  |
| **SignalRange** | Signal Range 8 bit (0-255) for angle |
| **Unit** | |  |

Table: Encoding Details of RgstrVertllbr\_An\_Actl

Register\_LHOB

Register location driver outboard

**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) |  |  |
| **Register\_LHOB\_Horizontal** |  |
| **Register\_LHOB\_Vertical** |  |
| **NoChange** | Logical signal for register LHOB no changel position |
| **Open** | Logical signal for register LHOB open position |
| **Close** | Logical signal for register LHOB close position |
| **Unit** | |  |

Table: Encoding Details of Register\_LHOB

RgstrSetObr\_D\_Rq

Change Aim Command for Righthand Intboard Register

**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) |  |  |
| **0x0** |  |
| **0x1** |  |
| **0x2** |  |
| **0x3** |  |
| **0x4** |  |
| **0x5** |  |
| **0x6** |  |
| **0x7** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrSetObr\_D\_Rq

SensorFeedBack\_LHOB\_Vertical

Register sensor position feedback to HMI for Left hand inboard register vertical voltage

**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 SensorFeedBack\_LHOB\_Vertical

APIM to RCCM Rgstr Set

**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) |  |  |
| **Register Set IBR** |  |
| **Register Set OBR** |  |
| **Register Set OBL** |  |
| **Register Set IBL** |  |
| **Unit** | |  |

Table: Encoding Details of APIM to RCCM Rgstr Set

RCCM to Actuator LIN

Signals sent from RCCM to register actuator

**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) |  |  |
| **ActCmdLHOBHoriz** | Implementation signal for ActCmdLHOBHoriz |
| **ActCmdLHOBVert** | Implementation signal for ActCmdLHOBVert |
| **ActCmdLHIBHoriz** | Implementation signal for ActCmdLHIBHoriz |
| **ActCmdLHIBVert** | Implementation signal for ActCmdLHIBVert |
| **ActCmdRHIBHoriz** | Implementation signal for ActCmdRHIBHoriz |
| **ActCmdRHIBVert** | Implementation signal for ActCmdRHIBVert |
| **ActCmdRHOBHoriz** | Implementation signal for ActCmdRHOBHoriz |
| **ActCmdRHOBVert** | Implementation signal for ActCmdRHOBVert |
| **LIN** |  |
| **Hardwire** |  |
| **Unit** | |  |

Table: Encoding Details of RCCM to Actuator LIN

ActuatorCommand\_RHOB\_Vertical

Right hand outboard register voltage for left right 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) |  |  |
| **Voltage** | Logical signal for RHOB vertical percentage change |
| **Unit** | |  |

Table: Encoding Details of ActuatorCommand\_RHOB\_Vertical

ActvButtnLeft\_D\_Stat

Subfeature Selection Status for Left Side Registers:

\*NonePressed

\*OnBodyLHS

\*OffBodyLHS

\*PresetOneLHS

\*PresetTwoLHS

\*Cycle\_8\_LHS

\*Cycle\_C\_LHS

\*Cycle\_O\_LHS

\*Cycle\_--\_LHS

\*Cycle\_I\_LHS

**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) |  |  |
| **0x0 nonePressed** | Button pressed discrete signal for none pressed LHS selection |
| **0x1 onBodyLHS** | Button pressed discrete signal for on body LHS selection |
| **0x2 offBodyLHS** | Button pressed discrete signal for off body LHS selection |
| **0x3 presetOneLHS** | Button pressed discrete signal for preset one LHS selection |
| **0x4 presetTwoLHS** | Button pressed discrete signal for preset two LHS selection |
| **0x6 presetTwoSaveLHS** | Button pressed discrete signal for preset two saved LHS selection |
| **0x5 presetOneSaveLHS** | Button pressed discrete signal for preset one saved LHS selection |
| **0x5 cycle\_8\_LHS** | Button pressed discrete signal for cycling figure 8 vanes LHS selection |
| **0x6 cycle\_C\_LHS** | Button pressed discrete signal for closed vanes LHS selection |
| **0x7 cycle\_O\_LHS** | Button pressed discrete signal for open vanes LHS selection |
| **0x8 cycle\_--\_LHS** | Button pressed discrete signal for cycling horizontal vanes LHS selection |
| **0x9 cycle\_|\_LHS** | Button pressed discrete signal for cycling vertical vanes LHS selection |
| **Unit** | |  |

Table: Encoding Details of ActvButtnLeft\_D\_Stat

RgstrVertlbr\_An\_Rq

Command Position of Vertical Vanes for Righthand Inboard Register

**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) |  |  |
| **unnamed1** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrVertlbr\_An\_Rq

HMIFeedback\_LHOB

Register position feedback to HMI

**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) |  |  |
| **Position** | Logical signal HMI feedback LHOB position |
| **Close** | Logical signal HMI feedback LHOB close |
| **Unit** | |  |

Table: Encoding Details of HMIFeedback\_LHOB

Register\_RHOB

Register location passenger outboard

**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) |  |  |
| **NoChange** | Logical signal for register RHOB no change position |
| **ManualChange** | Logical signal for register RHOB manual change position |
| **FeatureChange** | Logical signal for register RHOB feature change position |
| **Open** | Logical signal for register RHOB open position |
| **Close** | Logical signal for register RHOB close position |
| **Unit** | |  |

Table: Encoding Details of Register\_RHOB

RgstrHzntlIbl\_An\_Actl

Aim Status of Lefthand Inboard Register Horizontal Vanes

**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 RgstrHzntlIbl\_An\_Actl

RCCM to Actuator HRD

**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) |  |  |
| **ActCmdLHOBHoriz** |  |
| **Unit** | |  |

Table: Encoding Details of RCCM to Actuator HRD

Actuate Register Command

Signal requesting actuation of register position changes

**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) |  |  |
| **ActuatorCmnd\_LHOB\_Hor** |  |
| **actuatorCommand\_LHOB\_Horizontal** |  |
| **ActuatorCmnd\_LHOB\_Ver** |  |
| **ActuatorCmnd\_LHIB\_Hor** |  |
| **ActuatorCmnd\_LHIB\_Ver** |  |
| **ActuatorCmnd\_RHIB\_Hor** |  |
| **ActuatorCmnd\_RHIB\_Ver** |  |
| **ActuatorCmnd\_RHOB\_Hor** |  |
| **ActuatorCmnd\_RHOB\_Ver** |  |
| **Unit** | |  |

Table: Encoding Details of Actuate Register Command

ActuatorCommand\_RHIB\_Horizontal

Right hand inboard register voltage for up down 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) |  |  |
| **Voltage** | Logical signal for RHIB horizontal percentage change |
| **Unit** | |  |

Table: Encoding Details of ActuatorCommand\_RHIB\_Horizontal

Actuator to Sensor HRD

**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) |  |  |
| **SensedRegPosLHOBHoriz** |  |
| **SensedRegPosLHOBVert** |  |
| **SensedRegPosLHIBHorz** |  |
| **SensedRegPosLHIBVert** |  |
| **SensedRegPosRHOBHoriz** |  |
| **SensedRegPosRHOBVert** |  |
| **SensedRegPosRHIBHoriz** |  |
| **SensedRegPosRHIBVert** |  |
| **Unit** | |  |

Table: Encoding Details of Actuator to Sensor HRD

SensFdbkLHOBHoriz

Sense left hand horizontal outboard register vane positions and forward to RCCM

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of SensFdbkLHOBHoriz

ActCmdLHOBVert

Signal from RCCM to left hand outboard vertical vane register actuator

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of ActCmdLHOBVert

SensedRegPosRHOBhoriz

**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 SensedRegPosRHOBhoriz

HMIFeedback\_LHOB\_Horizontal

Register position feedback to HMI

**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 HMIFeedback\_LHOB\_Horizontal

ActCmdRHOBHoriz

Signal from RCCM to right hand outboard horizontal vane register actuator

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of ActCmdRHOBHoriz

RgstrHzntlObr\_An\_Actl

Aim Status of Righthand Outboard Register Horizontal Vanes

**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 RgstrHzntlObr\_An\_Actl

SensedRegPosLHOBhoriz

**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 SensedRegPosLHOBhoriz

RgstrVertObr\_An\_Actl

Aim Status of Righthand Outboard Register Vertical Vanes

**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) |  |  |
| **unnamed1** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrVertObr\_An\_Actl

HMIFeedback\_RHOB\_Vertical

Register position feedback to HMI

**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 HMIFeedback\_RHOB\_Vertical

ActuatorCommand\_RHIB\_Vertical

Right hand inboard register voltage for left right 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) |  |  |
| **Voltage** | Logical signal for RHIB vertical percentage change |
| **Unit** | |  |

Table: Encoding Details of ActuatorCommand\_RHIB\_Vertical

Requester Type

Signal to indicate if register move request is from another feature or user

**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) |  |  |
| **Unsupervised** |  |
| **Supervised** |  |
| **Unit** | |  |

Table: Encoding Details of Requester Type

Actuator to RCCM LIN

Signals sent from register actuator to RCCM

**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) |  |  |
| **SensFdbkLHOBHoriz** | Implementation signal for SensFdbkLHOBHoriz |
| **SensFdbkLHOBVert** | Implementation signal for SensFdbkLHOBVert |
| **SensFdbkLHIBHoriz** | Implementation signal for SensFdbkLHIBHoriz |
| **SensFdbkLHIBVert** | Implementation signal for SensFdbkLHIBVert |
| **SensFdbkRHIBHoriz** | Implementation signal for SensFdbkRHIBHoriz |
| **SensFdbkRHIBVert** | Implementation signal for SensFdbkRHIBVert |
| **SensFdbkRHOBHoriz** | Implementation signal for SensFdbkRHOBHoriz |
| **SensFdbkRHOBVert** | Implementation signal for SensFdbkRHOBVert |
| **Unit** | |  |

Table: Encoding Details of Actuator to RCCM LIN

HMIFeedback\_RHIB\_Horizontal

Register position feedback to HMI

**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 HMIFeedback\_RHIB\_Horizontal

SensedRegPosLHIBhoriz

**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 SensedRegPosLHIBhoriz

ActCmdRHIBVert

Signal from RCCM to right hand inboard vertical vane register actuator

**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) |  |  |
| **voltage** | voltage property |
| **Unit** | |  |

Table: Encoding Details of ActCmdRHIBVert

VehStat

Vehicle status signals used to store register positions when vehicle is on

**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) |  |  |
| **unnamed1** |  |
| **ON** | Implementation signal for vehicle status=on |
| **OFF** | Implementation signal for vehicle status=OFF |
| **ACCESSORY** | Implementation signal for vehicle status=ACCESSORY |
| **Unit** | |  |

Table: Encoding Details of VehStat

Register\_RHOB\_Vertical

Register airflow direction left or right

**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) |  |  |
| **Percentage** | Logical signal for register RHOB horizontal percentage change position vertical |
| **Unit** | |  |

Table: Encoding Details of Register\_RHOB\_Vertical

userFeedback

User feedback sent to the HMI visual output

**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) |  |  |
| **HMI Visual Output** | logical signal for HMI visual output |
| **Unit** | |  |

Table: Encoding Details of userFeedback

RgstrVertObl\_An\_Rq

Command Position of Vertical Vanes for Lefthand Outboard Register

**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) |  |  |
| **unnamed1** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrVertObl\_An\_Rq

SensorFeedback\_RHOB\_Horizontal

Register sensor position feedback to HMI for right hand outboard register horizontal voltage

**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 SensorFeedback\_RHOB\_Horizontal

Register\_Location

Register location in IP

**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) |  |  |
| **Register\_LHOB** |  |
| **Register\_LHIB** |  |
| **Register\_RHIB** |  |
| **Register\_RHOB** |  |
| **Unit** | |  |

Table: Encoding Details of Register\_Location

RCCM to APIM ActvButtn

Signals sent from RCCM to APIM

**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) |  |  |
| **ActiveButtonLHS\_D\_Stat** | Implementation signal from RCCM to APIM for ActiveButtonsLHS\_D\_Stat |
| **ActiveButtonRHS\_D\_Stat** | Implementation signal from RCCM to APIM for ActiveButtonsRHS\_D\_Stat |
| **Unit** | |  |

Table: Encoding Details of RCCM to APIM ActvButtn

ActuatorCommand\_RHOB\_Horizontal

Right hand outboard register voltage for up down 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) |  |  |
| **Voltage** | Logical signal for RHOB horizontal percentage change |
| **Unit** | |  |

Table: Encoding Details of ActuatorCommand\_RHOB\_Horizontal

SensedRegPosRHIBhoriz

**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 SensedRegPosRHIBhoriz

SensedRegPosLHOBvert

**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 SensedRegPosLHOBvert

HMIFeedback\_RHOB\_Horizontal

Register position feedback to HMI

**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 HMIFeedback\_RHOB\_Horizontal

HMIFeedback\_LHIB\_Horizontal

Register position feedback to HMI

**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 HMIFeedback\_LHIB\_Horizontal

HMIFeedback\_RHIB

Register position feedback to HMI

**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) |  |  |
| **Position** | Logical signal HMI feedback RHIB position |
| **Close** | Logical signal HMI feedback RHIB close |
| **Unit** | |  |

Table: Encoding Details of HMIFeedback\_RHIB

ActvButtnLeft\_D\_Rq

Subfeature Selection Commands for Left Side Registers

**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) |  |  |
| **0x0 = nonePressed (Default)** |  |
| **0x1 = onBodyRHS** |  |
| **0x2 = offBodyRHS** |  |
| **0x3 = presetOneLHS** |  |
| **0x4 = presetTwoRHS** |  |
| **0x5** |  |
| **0x6** |  |
| **0x7** |  |
| **0x8** |  |
| **0x9** |  |
| **0x10** |  |
| **0x11 = cycle\_|\_RHS** |  |
| **0xC - 0xF** |  |
| **Unit** | |  |

Table: Encoding Details of ActvButtnLeft\_D\_Rq

ActuatorCommand\_LHOB\_Vertical

Left hand outboard register voltage for up down 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) |  |  |
| **Voltage** | Logical signal for LHOB vertical percentage change |
| **Unit** | |  |

Table: Encoding Details of ActuatorCommand\_LHOB\_Vertical

Register\_LHIB\_Vertical

Register vane left or right airflow direction

**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) |  |  |
| **Percentage** | Logical signal for register LHIB horizontal percentage change position vertical |
| **Unit** | |  |

Table: Encoding Details of Register\_LHIB\_Vertical

SensorFeedback\_LHOB\_Horizontal

Register sensor position feedback to HMI for Left hand outboard register horizontal voltage

**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 SensorFeedback\_LHOB\_Horizontal

RgstrSetIbl\_D\_Rq

Change Aim Command for Lefthand Inboard Register

**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) |  |  |
| **0x0** |  |
| **0x1** |  |
| **0x2** |  |
| **0x3** |  |
| **0x4** |  |
| **0x5** |  |
| **0x6** |  |
| **0x7** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrSetIbl\_D\_Rq

Request HRD

Place holder for hardwire signals

**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 HRD

RgstrHzntlObl\_An\_Actl

Aim Status of Lefthand Outboard Register Horizontal Vanes

**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) |  |  |
| **unnamed1** |  |
| **Unit** | |  |

Table: Encoding Details of RgstrHzntlObl\_An\_Actl

SensorFeedBack\_RHIB\_Vertical

Register sensor position feedback to HMI for right hand inboard register vertical voltage

**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 SensorFeedBack\_RHIB\_Vertical

Register\_RHIB\_Vertical

Register airflow direction left or right

**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) |  |  |
| **Percentage** | Logical signal for register RHIB horizontal percentage change position vertical |
| **Unit** | |  |

Table: Encoding Details of Register\_RHIB\_Vertical

Register\_LHIB

Register location driver inboard

**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) |  |  |
| **NoChange** | Logical signal for register LHIB no changel position |
| **ManualChange** | Logical signal for register LHIB no change position |
| **FeatureChange** | Logical signal for register LHIB Feature change position |
| **Open** | Logical signal for register LHIB open position |
| **Close** | Logical signal for register LHIB close position |
| **Unit** | |  |

Table: Encoding Details of Register\_LHIB

ActuatorCommand\_LHIB\_Vertical

Register left hand inboard voltage for vertical 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) |  |  |
| **Voltage** | Logical signal for LHIB vertical percentage change |
| **Unit** | |  |

Table: Encoding Details of ActuatorCommand\_LHIB\_Vertical

EM\_PredefinedPositionsRequest

Move register commands based on register postion settings or button options

**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) |  |  |
| **ButtonPressedRHS** |  |
| **ButtonPressedLHS** |  |
| **requesterType** |  |
| **Unit** | |  |

Table: Encoding Details of EM\_PredefinedPositionsRequest

APIM to RCCM

Signals sent from APIM to RCCM

**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) |  |  |
| **RgstrHzntlIbl\_An\_Rq** |  |
| **RgstrHzntIbr\_An\_Rq** |  |
| **RgstrHzntlObl\_An\_Rq** |  |
| **RgstrHzntlObr\_An\_Rq** |  |
| **RgstrVertIbl\_An\_Rq** |  |
| **RgstrVertbr\_An\_Rq** |  |
| **RgstrVertObl\_An\_Rq** |  |
| **RgstrVertObr\_An\_Rq** |  |
| **Apim to Rccm Buttons** |  |
| **APIM to RCCM RGSTR SET** |  |
| **Unit** | |  |

Table: Encoding Details of APIM to RCCM

Clmt\_Button\_Stat4

Climate button 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) |  |  |
| **None\_Pressed** | Implementation signal for climate button none pressed |
| **Front\_Power\_Pressed** | Implementation signal for climate button front power pressed |
| **Unit** | |  |

Table: Encoding Details of Clmt\_Button\_Stat4

ActvButtnRight\_D\_Rq

Subfeature Selection Commands for Right Side Registers

**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) |  |  |
| **0x0 = nonePressed (Default)** |  |
| **0x1 = onBodyRHS** |  |
| **0x2 = offBodyRHS** |  |
| **0x3 = presetOneLHS** |  |
| **0x4 = presetTwoRHS** |  |
| **0x5** |  |
| **0x6** |  |
| **0x7** |  |
| **0x8** |  |
| **0x9** |  |
| **0x10** |  |
| **0x11 = cycle\_|\_RHS** |  |
| **0xC - 0xF** |  |
| **Unit** | |  |

Table: Encoding Details of ActvButtnRight\_D\_Rq

RegHomePositions

Signal for register home positions

**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) |  |  |
| **Reg\_LHIB\_Horz\_Home** |  |
| **Reg\_LHIB\_Vert\_Home** |  |
| **Reg\_LHOB\_Horz\_Home** |  |
| **Reg\_LHOB\_Vert\_Home** |  |
| **Reg\_RHIB\_Horz\_Home** |  |
| **Reg\_RHIB\_Vert\_Home** |  |
| **Reg\_RHOB\_Horz\_Home** |  |
| **Reg\_RHOB\_Vert\_Home** |  |
| **Reg\_LHIB\_Horz\_Angle\_Percentage** |  |
| **Reg\_LHIB\_Vert\_Angle\_Percentage** |  |
| **Reg\_LHOB\_Horz\_Angle\_Percentage** |  |
| **Reg\_LHOB\_Vert\_Angle\_Percentage** |  |
| **Reg\_RHIB\_Horz\_Angle\_Percentage** |  |
| **Reg\_RHOB\_Vert\_Angle\_Percentage** |  |
| **Reg\_RHIB\_Vert\_Angle\_Percentage** |  |
| **Reg\_RHOB\_Horz\_Angle\_Percentage** |  |
| **Unit** | |  |

Table: Encoding Details of RegHomePositions

Register\_LHOB\_Vertical

Register airflow left or right airflow direction

**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) |  |  |
| **Percentage** | Logical signal for register LHOB horizontal percentage change position vertical |
| **Unit** | |  |

Table: Encoding Details of Register\_LHOB\_Vertical

RgstrHzntlObr\_An\_Rq

Command Position of Horizontal Vanes for Righthand Outboard Register

**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 RgstrHzntlObr\_An\_Rq

RCCM to APIM

**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) |  |  |
| **Register Horz IBL** |  |
| **Register Horz IBR** |  |
| **Register Horz OBL** |  |
| **Register Horz OBR** |  |
| **Register Vert IBL** |  |
| **Register Vert IBR** |  |
| **Register Vert OBL** |  |
| **Register Vert OBR** |  |
| **Active Buttons** |  |
| **Unit** | |  |

Table: Encoding Details of RCCM to APIM

vehicleStatus

Vehicle on off status used by register move commands

**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) |  |  |
| **ON\_TO\_OFF** | vehicle status on |
| **OFF\_TO\_ON** | vehicle status off |
| **Accessory** | vehicle status accessory |
| **Unit** | |  |

Table: Encoding Details of vehicleStatus

ActvButtnRight\_D\_Stat

Subfeature Selection Status for Right Side Registers:

\*NonePressed

\*OnBodyLHS

\*OffBodyLHS

\*PresetOneLHS

\*PresetTwoLHS

\*Cycle\_8\_LHS

\*Cycle\_C\_LHS

\*Cycle\_O\_LHS

\*Cycle\_--\_LHS

\*Cycle\_I\_LHS

**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) |  |  |
| **0x0 nonePressed** | Button pressed discrete signal for none pressed RHS selection |
| **0x1 onBodyRHS** | Button pressed discrete signal for on body RHS selection |
| **0x2 offBodyRHS** | Button pressed discrete signal for off body RHS selection |
| **0x3 presetOneRHS** | Button pressed discrete signal for preset one RHS selection |
| **0x4 presetTwoRHS** | Button pressed discrete signal for preset two RHS selection |
| **0x6 presetTwoSaveRHS** | Button pressed discrete signal for preset one saved RHS selection |
| **0x5 cycle\_8\_RHS** | Button pressed discrete signal cycling in figure 8 RHS selection |
| **0x5 presetOneSaveRHS** | Button pressed discrete signal for preset one saved RHS selection |
| **0x6 cycle\_C\_RHS** | Button pressed discrete signal for closed vanes RHS selection |
| **0x7 cycle\_O\_RHS** | Button pressed discrete signal open vanes RHS selection |
| **0x8 cycle\_--\_RHS** | Button pressed discrete signal for cycling vanes in horizontal direction RHS selection |
| **0x9 cycle\_|\_RHS** | Button pressed discrete signal for cycling vanes in vertical direction RHS selection |
| **Unit** | |  |

Table: Encoding Details of ActvButtnRight\_D\_Stat

RegisterPositionHMIFeedback

Register sensor position feedback to HMI

**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) |  |  |
| **unnamed1** |  |
| **HMIFeedbck\_LHOB\_Horz** |  |
| **HMIFeedbck\_LHOB\_Vert** |  |
| **HMIFeedbck\_LHIB\_Horz** |  |
| **HMIFeedbck\_LHIB\_Vert** |  |
| **HMIFeedbck\_RHIB\_Horz** |  |
| **HMIFeedbck\_RHIB\_Vert** |  |
| **HMIFeedbck\_RHOB\_Horz** |  |
| **HMIFeedbck\_RHOB\_Vert** |  |
| **Unit** | |  |

Table: Encoding Details of RegisterPositionHMIFeedback

RgstrHzntlIbl\_An\_Rq

Command Position of Horizontal Vanes for Lefthand Inboard Register

**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 RgstrHzntlIbl\_An\_Rq

MoveAirFlowCommand

Move register commands from user input or another feature

**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) |  |  |
| **userInput** | Logical signal for move airflow command for user input to move registers |
| **Unit** | |  |

Table: Encoding Details of MoveAirFlowCommand

HMIFeedback\_LHIB

Register position feedback to HMI

**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) |  |  |
| **Position** | Logical signal HMI feedback LHIB position |
| **Close** | Logical signal HMI feedback LHIB close |
| **Unit** | |  |

Table: Encoding Details of HMIFeedback\_LHIB

HMIFeedback\_LHOB\_Vertical

Register position feedback to HMI

**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 HMIFeedback\_LHOB\_Vertical

Document ends here.