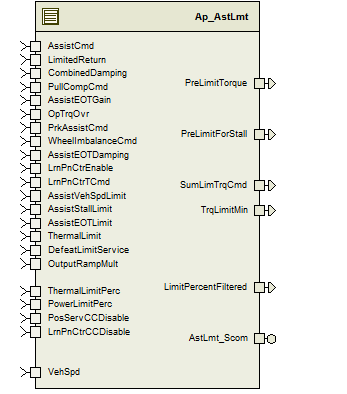
# Module – Assist Sum and Limit (Current Mode)

# High-Level Description

This module combines and limits the various assist command signals from EPS modules. It puts out several torque commands from different points in the summation and limiting process.

# Figures

## Component Diagram



# Variable Data Dictionary

For details on module input / output variable, refer to the Data Dictionary for the application. Input / output variable names are listed here for reference.

|  |  |  |
| --- | --- | --- |
| Module Inputs | Module Outputs | |
| AssistCmd\_MtrNm\_f32 | | LimitPercentFiltered\_Uls\_f32 |
| AssistEOTDamping\_MtrNm\_f32 | |  |
| AssistEOTGain\_Uls\_f32 | |  |
| AssistEOTLimit\_MtrNm\_f32 | | PreLimitForStall\_MtrNm\_f32 |
|  | | PreLimitTorque\_MtrNm\_f32 |
| AssistStallLimit\_MtrNm\_f32 | | SumLimTrqCmd\_MtrNm\_T\_f32 |
| AssistVehSpdLimit\_MtrNm\_f32 | | TrqLimitMin\_MtrNm\_f32 |
| CombinedDamping\_MtrNm\_f32 | |  |
| DefeatLimitService\_Cnt\_lgc | |  |
|  | |  |
| LimitedReturn\_MtrNm\_f32 | |  |
| LrnPnCtrCCDisable\_Cnt\_lgc | |  |
| LrnPnCtrEnable\_Cnt\_lgc | |  |
| LrnPnCtrTCmd\_MtrNm\_f32 | |  |
|  | |  |
| OpTrqOvr\_MtrNm\_f32 | |  |
| OutputRampMult\_Uls\_f32 | |  |
| PosServCCDisable\_Cnt\_lgc | |  |
| PowerLimitPerc\_Uls\_f32 | |  |
| PrkAssistCmd\_MtrNm\_f32 | |  |
| PullCompCmd\_MtrNm\_f32 | |  |
| ThermalLimitPerc\_Uls\_f32 | |  |
| ThermalLimit\_MtrNm\_f32 | |  |
| VehSpd\_Kph\_f32 | |  |
| WheelImbalanceCmd\_MtrNm\_f32 | |  |

## Module Internal Variables

This section identifies the name, range and resolutions for module specific data created by this module. If there are no range restrictions on the variable, the term “FULL” is placed into the table for legal range.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Variable Name | Resolution | Legal Range  (min) | Legal Range  (max) | Software Segment |
| AstLmt\_ManualTrqCmd\_MtrNm\_M\_f32 | Single Precision Float | -8.8 | 8.8 | ASTLMT\_START\_SEC\_VAR\_CLEARED\_32 |
| AstLmt\_ManualTrqCmdEn\_Cnt\_M\_lgc | n/a | FALSE | TRUE | ASTLMT\_START\_SEC\_VAR\_CLEARED\_BOOLEAN |
| AstLmt\_SteeringAsstDefeat\_Cnt\_M\_lgc | n/a | FALSE | TRUE | ASTLMT\_START\_SEC\_VAR\_CLEARED\_BOOLEAN |

### User defined typedef definition/declaration

This section documents any user types uniquely used for the module.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Typedef Name | Element Name | User Defined Type | Legal Range  (min) | Legal Range  (max) |
| None |  |  |  |  |

# Constant Data Dictionary

## Calibration Constants

This section lists the calibrations used by the module. For details on calibration constants, refer to the Data Dictionary for the application.

|  |
| --- |
| Constant Name |
| k\_SumLimPlCmpLimit\_MtrNm\_f32 |
|  |

## Program(fixed) Constants

### Embedded Constants

All embedded constants whose values are provided in Eng units will be evaluated to the equivalent counts by using the FPM\_InitFixedPoint\_m() macro within the #define statement.

#### Local

|  |  |  |  |
| --- | --- | --- | --- |
| Constant Name | Resolution | Units | Value |
| None |  |  |  |

#### Global

This section lists the global constants used by the module. For details on global constants, refer to the Data Dictionary for the application.

|  |
| --- |
| Constant Name |
| D\_ZERO\_ULS\_F32 |
| D\_ZERO\_CNT\_S8 |
| D\_ONE\_ULS\_F32 |
| D\_MTRTRQCMDLOLMT\_MTRNM\_F32 |
| D\_MTRTRQCMDHILMT\_MTRNM\_F32 |
| FLT\_EPSILON |

### Module specific Lookup Tables Constants

|  |  |  |  |
| --- | --- | --- | --- |
| Constant Name | Resolution | Value | Software Segment |
| None |  |  |  |

# Functions/Macros used by the Sub-Modules

## Library Functions / Macros

The library and functions / Macros that are called by the various sub modules are identified below,

1. Abs\_f32\_m
2. Sign\_f32\_m
3. Min\_m
4. Max\_m
5. Limit\_m

## Data Hiding Functions

1. Rte\_Call\_SteeringAsstDefeat\_WriteBlock
2. Rte\_Pim\_SteerAsstDefeat

## Global Functions/Macros Defined by this Module

None

## Local Functions/Macros Used by this MDD only

None

# Software Module Implementation

## Runtime Environment (RTE) Initial Values

This section lists the initial values of data written by this module but controlled by the RTE. After RTE initialization, the data in this table will contain these values.

|  |  |
| --- | --- |
| Data | Value |
| Rte\_InitValue\_AssistCmd\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_AssistEOTDamping\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_AssistEOTGain\_Uls\_f32 | 1 |
| Rte\_InitValue\_AssistEOTLimit\_MtrNm\_f32 | 8.8 |
| Rte\_InitValue\_AssistStallLimit\_MtrNm\_f32 | 8.8 |
| Rte\_InitValue\_AssistVehSpdLimit\_MtrNm\_f32 | 8.8 |
| Rte\_InitValue\_CombinedDamping\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_DefeatLimitService\_Cnt\_lgc | FALSE |
| Rte\_InitValue\_LimitPercentFiltered\_Uls\_f32 | 0 |
| Rte\_InitValue\_LimitedReturn\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_LrnPnCtrCCDisable\_Cnt\_lgc | FALSE |
| Rte\_InitValue\_LrnPnCtrEnable\_Cnt\_lgc | FALSE |
| Rte\_InitValue\_LrnPnCtrTCmd\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_OpTrqOvr\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_OutputRampMult\_Uls\_f32 | 0 |
| Rte\_InitValue\_PosServCCDisable\_Cnt\_lgc | FALSE |
| Rte\_InitValue\_PowerLimitPerc\_Uls\_f32 | 0 |
| Rte\_InitValue\_SumLimTrqCmd\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_PreLimitForStall\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_PreLimitTorque\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_PrkAssistCmd\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_PullCompCmd\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_ThermalLimit\_MtrNm\_f32 | 8.8 |
| Rte\_InitValue\_ThermalLimitPerc\_Uls\_f32 | 0 |
| Rte\_InitValue\_VehSpd\_Kph\_f32 | 0 |
| Rte\_InitValue\_WheelImbalanceCmd\_MtrNm\_f32 | 0 |

## Initialization Functions

### Init: AstLmt\_Init

#### Design Rationale

#### Program Flow Start

N/A

#### Store Module Inputs to Local Copies

N/A

#### (Processing of function)…..

AstLmt\_SteeringAsstDefeat\_Cnt\_M\_lgc = \*Rte\_Pim\_SteerAsstDefeat()

#### Store Local copy of outputs into Module Outputs

N/A

#### Program Flow End

N/A

## Periodic Functions

### Per: AstLmt\_Per1

#### Design Rationale

While the FDD specifies the LimitPercentFiltered output to be populated every 10 ms, the overhead required for another periodic function would be greater than including the single Max\_m() macro in the main 2 ms periodic function.

#### Program Flow Start

N/A

#### Store Module Inputs to Local copies

AssistCmd\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_AssistCmd\_MtrNm\_f32()

AssistEOTDamping\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_AssistEOTDamping\_MtrNm\_f32()

AssistEOTGain\_Uls\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_AssistEOTGain\_Uls\_f32()

AssistEOTLimit\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_AssistEOTLimit\_MtrNm\_f32()

AssistStallLimit\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_AssistStallLimit\_MtrNm\_f32()

AssistVehSpdLimit\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_AssistVehSpdLimit\_MtrNm\_f32()

CombinedDamping\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_CombinedDamping\_MtrNm\_f32()

DefeatLimitService\_Cnt\_T\_lgc = Rte\_IRead\_AstLmt\_Per1\_DefeatLimitService\_Cnt\_lgc()

LimitedReturn\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_LimitedReturn\_MtrNm\_f32()

LrnPnCtrCCDisable\_Cnt\_T\_lgc = Rte\_IRead\_AstLmt\_Per1\_LrnPnCtrCCDisable\_Cnt\_lgc()

LrnPnCtrEnable\_Cnt\_T\_lgc = Rte\_IRead\_AstLmt\_Per1\_LrnPnCtrEnable\_Cnt\_lgc()

LrnPnCtrTCmd\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_LrnPnCtrTCmd\_MtrNm\_f32()

OpTrqOvr\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_OpTrqOvr\_MtrNm\_f32()

OutputRampMult\_Uls\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_OutputRampMult\_Uls\_f32()

PosServCCDisable\_Cnt\_T\_lgc = Rte\_IRead\_AstLmt\_Per1\_PosServCCDisable\_Cnt\_lgc()

PowerLimitPerc\_Uls\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_PowerLimitPerc\_Uls\_f32()

PrkAssistCmd\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_PrkAssistCmd\_MtrNm\_f32()

PullCompCmd\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_PullCompCmd\_MtrNm\_f32()

ThermalLimitPerc\_Uls\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_ThermalLimitPerc\_Uls\_f32()

ThermalLimit\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_ThermalLimit\_MtrNm\_f32()

VehSpd\_Kph\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_VehSpd\_Kph\_f32()

WheelImbalanceCmd\_MtrNm\_T\_f32 = Rte\_IRead\_AstLmt\_Per1\_WheelImbalanceCmd\_MtrNm\_f32()

#### Preconditioning and Command summation



#### Apply Gain and Limit

#### Assist Reduction Level



#### Store Local copy of outputs into Module Outputs

Rte\_IWrite\_AstLmt\_Per1\_LimitPercentFiltered\_Uls\_f32(LimitPercentFiltered\_Uls\_T\_f32)

Rte\_IWrite\_AstLmt\_Per1\_TrqLimitMin\_MtrNm\_f32(TrqLimitMin\_MtrNm\_T\_f32);

Rte\_IWrite\_AstLmt\_Per1\_PostLimitForAssistSumCC\_MtrNm\_f32(PostLimitForAssistSumCC\_MtrNm\_T\_f32)

Rte\_IWrite\_AstLmt\_Per1\_SumLimTrqCmd\_MtrNm\_f32(SumLimTrqCmd\_MtrNm\_T\_f32)

Rte\_IWrite\_AstLmt\_Per1\_PreLimitForStall\_MtrNm\_f32(PreLimitForStall\_MtrNm\_T\_f32)

Rte\_IWrite\_AstLmt\_Per1\_PreLimitTorque\_MtrNm\_f32(PreLimitTorque\_MtrNm\_T\_f32)

#### Program Flow End

N/A

## Fault Recovery Functions

None

## Shutdown Functions

None

## Interrupt Functions

None

## Serial Communication Functions

### SCom: AstLmt\_SCom\_ManualTrqCmd

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Type | Min | Max | UTP Tol. |
| **Arguments Passed** | EnableManualCtrl | boolean | FALSE | TRUE |  |
|  | MtrTrqCmd\_MtrNm\_f32 | float32 | -16 | 15.9995 |  |
| **Return Value** | RetCode | Std\_ReturnType | 0 | 34 | 0 |

#### Design Rationale

None

#### Program Flow Start

N/A

#### Store Module Inputs to Local copies

Rte\_Read\_VehSpd\_Kph\_f32(&VehSpd\_Kph\_T\_f32)

#### Process Manual Torque Command



#### Store Local copy of outputs into Module Outputs

None

#### Program Flow End

N/A

### SCom: AstLmt\_Scom\_GetSteeringAssistDefeat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Type | Min | Max | UTP Tol. |
| **Arguments Passed** | SteeringAsstDefeat\_Cnt\_lgc | \*boolean | FALSE | TRUE |  |
| **Return Value** | none |  |  |  |  |

#### Design Rationale

None

#### Program Flow Start

N/A

#### Store Module Inputs to Local copies

N/A

#### Get Steering Assist Defeat Status

\*SteeringAsstDefeat\_Cnt\_lgc = \*Rte\_Pim\_SteerAsstDefeat()

#### Store Local copy of outputs into Module Outputs

None

#### Program Flow End

N/A

### SCom: AstLmt\_Scom\_SetSteeringAssistDefeat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Type | Min | Max | UTP Tol. |
| **Arguments Passed** | SteeringAsstDefeat\_Cnt\_lgc | boolean | FALSE | TRUE |  |
| **Return Value** | none |  |  |  |  |

#### Design Rationale

None

#### Program Flow Start

N/A

#### Store Module Inputs to Local copies

N/A

#### Get Steering Assist Defeat Status

\*Rte\_Pim\_SteerAsstDefeat() = SteeringAsstDefeat\_Cnt\_lgc

Rte\_Call\_SteeringAsstDefeat\_WriteBlock(NULL\_PTR)

#### Store Local copy of outputs into Module Outputs

None

#### Program Flow End

N/A

# Execution Requirements

## Execution Rates for sub-modules called by the Scheduler

This table serves as reference for the Scheduler design

|  |  |  |
| --- | --- | --- |
| Function Name | Calling Frequency | System State(s) in which the function is called |
| AstLmt\_Init | Executed Once after RTE is started | ColdInit |
| AstLmt\_Per1 | 2 ms | ALL |

## Execution Requirements for Serial Communication Functions

|  |  |
| --- | --- |
| Function Name | Sub-Module called by (Serial Comm Function Name) |
| AstLmt\_Scom\_ManualTrqCmd |  |
| AstLmt\_Scom\_GetSteeringAssistDefeat |  |
| AstLmt\_Scom\_SetSteeringAssistDefeat |  |

# Memory Map Definition Requirements

## Sub Modules (Functions)

This table identifies the software segments for functions identified in this module.

|  |  |
| --- | --- |
| Name of Sub Module | Software Segment |
| AstLmt\_Init | RTE\_START\_SEC\_AP\_ASTLMT\_APPL\_CODE |
| AstLmt\_Per1 | RTE\_START\_SEC\_AP\_ASTLMT\_APPL\_CODE |
| AstLmt\_Scom\_ManualTrqCmd | RTE\_START\_SEC\_AP\_ASTLMT\_APPL\_CODE |
| AstLmt\_Scom\_GetSteeringAssistDefeat | RTE\_START\_SEC\_AP\_ASTLMT\_APPL\_CODE |
| AstLmt\_Scom\_SetSteeringAssistDefeat | RTE\_START\_SEC\_AP\_ASTLMT\_APPL\_CODE |

## Local Functions

This table identifies the software segments for local functions identified in this module.

|  |  |
| --- | --- |
| Name of Sub Module | Software Segment |
| None |  |

# Known Issues / Limitations With Design

1. INLINE functions defined in GlobalMacro.h are not unit tested.

# Revision Control Log

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item #** | **Rev #** | **Change Description** | **Date** | **Author Initials** |
| 1 | 1.0 | Initial Version (SF-04B v001) | 07-Aug-12 | OT |
| 2 | 2.0 | Fixed UTP Issues (global constants) | 08-Aug-12 | OT |
| 3 | 3.0 | Added ManualTrqCmd service | 16-Aug-12 | OT |
| 4 | 4.0 | Replaced HwtrqPolarity with assistassembley polarity | 11-SEP-12 | SAH |
| 5 | 5.0 | - Removed Inputs: MRFMtrVel, AssistAssembly\_Polarity, Assist\_PowerLimit  - Removed Output: PostLimit\_ForAssistSumCC  - Renamed Output: PreLimit\_for\_Power to SumLimTrqCmd\_MtrNm  - Removed calibration: k\_OvrSpdMtrTrq2QLmt\_MtrNm | 01-Dec-12 | Selva |
| 6 | 6.0 | Updated output limit on sumlimtrqcmd from 0 to -8.8 to match FDD data dictionary | 14-Jan-13 | SAH |
| 7 | 7.0 | Updates to add steering assist defeat | 03-Jun-13 | VK |
| 8 | 8.0 | Update to v4 of FDD. Added new outputs and matched the naming conventions | 23-Nov-13 | Selva |