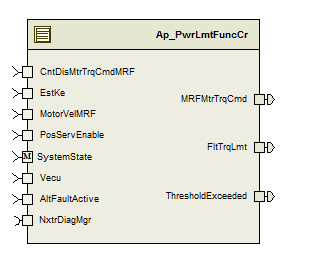
# Module –

# High-Level Description

This module determines an appropriate limit for the system motor torque command based on reasonable output power and system temperature. It also determines to what degree the system command is being limited.

# 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 | |
| EstKe\_VpRadpS\_f32 | | MRFMtrTrqCmd\_MtrNm\_f32 |
| MotorVelMRF\_MtrRadpS\_f32 | | FltTrqLmt\_Uls\_f32 |
| PosServEnable\_Cnt\_lgc | | ThresholdExceeded\_Cnt\_lgc |
| Vecu\_Volt\_f32 | |  |
| CntDisMtrTrqCmdMRF\_MtrNm\_f32 | |  |
| AltFaultActive\_Cnt\_lgc | |  |

## 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 |
| PwrLmtFuncCr\_ SpdAdj\_MtrRadpS\_M\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ VoltageRecoveryTimer\_mS\_M\_u32 | 1 | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ ThresholdExceeded\_Cnt\_M\_lgc | N/A | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_BOOLEAN |
| PwrLmtFuncCr\_ TrqLmtKSV\_M\_str | LPF32KSV\_Str | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_UNSPECIFIED |
| PwrLmtFuncCr\_ TrqLmtKSV\_M\_str.SV\_Uls\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary |  |
| PwrLmtFuncCr\_ TrqLmtKSV\_M\_str.K\_Uls\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary |  |
| PwrLmtFuncCr\_ MtrVelKSV\_M\_str | LPF32KSV\_Str | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_UNSPECIFIED |
| PwrLmtFuncCr\_ MtrVelKSV\_M\_str.SV\_Uls\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary |  |
| PwrLmtFuncCr\_ MtrVelKSV\_M\_str.K\_Uls\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary |  |
| PwrLmtFuncCr\_ MtrEnvSpd\_MtrRadpS\_M\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ MinStdOpLmt\_MtrNm\_M\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ TrqEnvLmt1\_MtrNm\_M\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ TrqEnvLmt4\_MtrNm\_D\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ TrqLmt4\_MtrNm\_D\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ OPVelOffset\_MtrRadpS\_D\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ TrqLmt1\_MtrNm\_D\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ TLimitMaxCurr\_MtrNm\_D\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ MinStdOpLmt\_MtrNm\_D\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ LimitDifference\_MtrNm\_D\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_TrqLmt\_Uls\_D\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ MtrVelFilt\_MtrRadpS\_D\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |
| PwrLmtFuncCr\_ VecuSlewAdj\_Volt\_M\_f32 | Single Precision Float | See Data Dictionary | See Data Dictionary | PWRLMTFUNCCR\_START\_SEC\_VAR\_CLEARED\_32 |

### 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) |
|  |  |  |  |  |

# 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 |
| t\_DLVTblX\_Volt\_u5p11[] |
| t\_DLVTblY\_MtrRadpS\_u11p5[] |
| t\_MtrEnvTblX\_MtrRadpS\_s11p4[] |
| t\_MtrEnvTblY\_MtrNm\_u5p11 |
| k\_KeStdTemp\_VpRadpS\_f32 |
| t\_StdOpMtrEnvTblX\_MtrRadpS\_u11p5[] |
| t\_StdOpMtrEnvTblY\_MtrNm\_u4p12[] |
| k\_SpdAdjSlewInc\_MtrRadpS\_f32 |
| k\_SpdAdjSlewDec\_MtrRadpS\_f32 |
| k\_SpdAdjSlewEnable\_Cnt\_lgc |
| k\_AsstReducLPFKn\_Hz\_f32 |
| k\_PwrLmtMtrVelLPFKn\_Hz\_f32 |
| k\_FiltAsstReducTh\_Uls\_f32 |
| k\_LowVltAstRecTime\_mS\_u16 |
| k\_LowVltAstRecTh\_Volt\_f32 |
| k\_PwrLmtVecuAltFltAdj\_Volt\_f32 |
| k\_PwrLmtVecuAdjSlew\_VoltspL\_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 |
| D\_10MS\_SEC\_F32 | Single Precision Floating Point | Sec | 0.010 |
| D\_FLTTRQLMTLOLMT\_F32 | Single Precision Floating Point | Unitless | 0.0 |
| D\_FLTTRQLMTHILMT\_F32 | Single Precision Floating Point | Unitless | 1.0 |

#### 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\_2MS\_SEC\_F32 |
| D\_ZERO\_ULS\_F32 |
| FLT\_EPSILON |
| D\_TESTNOTCOMPLETETHISOPCYCLEBIT\_CNT\_B8 |
| D\_MTRTRQCMDLOLMT\_MTRNM\_F32 |
| D\_MTRTRQCMDHILMT\_MTRNM\_F32 |
| D\_VECUMIN\_VOLTS\_F32 |
| D\_FALSE\_CNT\_LGC |

### 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. LPF\_KUpdate\_f32\_m
2. LPF\_OpUpdate\_f32\_m
3. Abs\_f32\_m
4. Abs\_s16\_m
5. FPM\_FixedToFloat\_m
6. FPM\_FloatToFixed\_m
7. IntplVarXY\_u16\_u16Xu16Y\_Cnt
8. TableSize\_m
9. Limit\_m
10. Min\_m
11. Max\_m
12. IntplVarXY\_u16\_s16Xu16Y\_Cnt
13. Sign\_f32\_m

## Data Hiding Functions

1. None

## 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 \_CntDisMtrTrqCmdMRF\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_EstKe\_VpRadpS\_f32 | 0 |
| Rte\_InitValue\_MtrVel\_MtrRadpS\_f32 | 0 |
| Rte\_InitValue\_PosServEnable\_Cnt\_lgc | FALSE |
| Rte\_InitValue\_FltTrqLmt\_Uls\_f32 | 0 |
| Rte\_InitValue\_MRFMtrTrqCmd\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_ThresholdExceeded\_Cnt\_lgc | FALSE |
| Rte\_InitValue\_Vecu\_Volt\_f32 | 5 |
| Rte\_InitValue\_AltFaultActive\_Cnt\_lgc | FALSE |

## Initialization Functions

### Init: \_Init1

#### Design Rationale

None

#### Module Outputs

None

#### Module Internal



## Periodic Functions

### Per: \_Per1

#### Design Rationale

None

#### Program Flow Start

Rte\_Call\_PwrLmtFuncCr\_Per1\_CP0\_CheckpointReached

#### Store Module Inputs to Local copies

EstKe\_VpRadpS\_T\_f32 = Rte\_IRead\_PwrLmtFuncCr\_Per1\_EstKe\_VpRadpS\_f32()

MotorVelMRF\_MtrRadpS\_T\_f32 = Rte\_IRead\_PwrLmtFuncCr\_Per1\_MotorVelMRF\_MtrRadpS\_f32()Vecu\_Volt\_T\_f32 = Rte\_IRead\_PwrLmtFuncCr\_Per1\_Vecu\_Volt\_f32()

CntDisMtrCmdMRF\_MtrNm\_T\_f32 = Rte\_IRead\_PwrLmtFuncCr\_Per1\_CntDisMtrTrqCmdMRF\_MtrNm\_f32()

AltFaultActive\_Cnt\_T\_lgc = Rte\_IRead\_PwrLmtFuncCr\_Per1\_AltFaultActive\_Cnt\_lgc()

#### Filter Motor Velocity

MtrVelFilt\_MtrRadpS\_T\_f32 = LPF\_OpUpdate\_f32\_m(MotorVelMRF\_MtrRadpS\_T\_f32, & PwrLmtFuncCr\_ MtrVelKSV\_M\_str)

#### Nexteer Power Limit



#### Output Velocity



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

PwrLmtFuncCr\_OPVelOffset\_MtrRadpS\_D\_f32 = OPVelOffset\_MtrRadpS\_T\_f32

PwrLmtFuncCr\_TrqEnvLmt\_MtrRadpS\_D\_f32 = TrqEnvLmt\_MtrRadpS\_T\_f32

PwrLmtFuncCr\_TLimitMaxCurr\_MtrNm\_D\_f32 = TLimitMaxCurr\_MtrNm\_T\_f32

PwrLmtFuncCr\_MinStdOpLmt\_MtrNm\_D\_f32 = PwrLmtFuncCr\_ MinStdOpLmt\_MtrNm\_M\_f32

PwrLmtFuncCr\_ SpdAdj\_MtrRadpS\_M\_f32 = SpdAdj\_MtrRadpS\_T\_f32

PwrLmtFuncCr\_ TrqEnvLmt1\_MtrNm\_M\_f32 =TrqEnvLmt1\_MtrNm\_T\_f32;

PwrLmtFuncCr\_TrqLmt1\_MtrNm\_D\_f32 = TrqLmt1\_MtrNm\_T\_f32;

PwrLmtFuncCr\_TrqEnvLmt4\_MtrNm\_D\_f32 = TrqEnvLmt4\_MtrNm\_T\_f32;

PwrLmtFuncCr\_TrqLmt4\_MtrNm\_D\_f32 = TrqLmt4\_MtrNm\_T\_f32;

PwrLmtFuncCr\_MtrVelFilt\_MtrRadpS\_D\_f32 = MtrVelFilt\_MtrRadpS\_T\_f32;

PwrLmtFuncCr\_ VecuSlewAdj\_Volt\_M\_f32 = PwrLmtVecu1SlewAdj\_Volt\_T\_f32

Rte\_IWrite\_PwrLmtFuncCr\_Per1\_MRFMtrTrqCmd\_MtrNm\_f32 (MRFMtrTrq\_MtrNm\_T\_f32)

#### Program Flow End

Rte\_Call\_PwrLmtFuncCr\_Per1\_CP1\_CheckpointReached

### Per: \_Per2

#### Design Rationale

None

#### Program Flow Start

Rte\_Call\_PwrLmtFuncCr\_Per2\_CP0\_CheckpointReached

#### Store Module Inputs to Local copies

CntDisMtrCmdMRF\_MtrNm\_T\_f32 = Rte\_IRead\_PwrLmtFuncCr\_Per2\_CntDisMtrTrqCmdMRF\_MtrNm\_f32();

Vecu\_Volt\_T\_f32 = Rte\_IRead\_PwrLmtFuncCr\_Per2\_Vecu\_Volt\_f32();

MinStdOpLmt\_MtrNm\_T\_f32 = PwrLmtFuncCr\_ MinStdOpLmt\_MtrNm\_M\_f32

TrqEnvLmt1\_MtrNm\_T\_f32 = PwrLmtFuncCr\_ TrqEnvLmt1\_MtrNm\_M\_f32

MtrEnvSpd\_MtrRadpS\_T\_f32 = PwrLmtFuncCr\_ MtrEnvSpd\_MtrRadpS\_M\_f32

#### Power Limit Status



#### Assist Limit Condition

#### 

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

PwrLmtFuncCr\_LimitDifference\_MtrNm\_D\_f32 = LimitDifference\_MtrNm\_T\_f32

PwrLmtFuncCr\_TrqLmt\_Uls\_D\_f32 = TrqLmt\_Uls\_T\_f32

Rte\_IWrite\_PwrLmtFuncCr\_Per2\_FltTrqLmt\_Uls\_f32(FltTrqLmt\_Uls\_T\_f32)

Rte\_IWrite\_PwrLmtFuncCr\_Per2\_ThresholdExceeded\_Cnt\_lgc(PwrLmtFuncCr\_ ThresholdExceeded\_Cnt\_M\_lgc)

#### Program Flow End

Rte\_Call\_PwrLmtFuncCr\_Per2\_CP1\_CheckpointReached

## Fault Recovery Functions

None

## Shutdown Functions

None

## Interrupt Functions

None

## Serial Communication Functions

None

# 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 |
| PwrLmtFuncCr\_Init1 | On Event | On Init |
| PwrLmtFuncCr\_Per1 | 2 ms | OPERATE |
| PwrLmtFuncCr\_Per2 | 2 ms | OPERATE |

## Execution Requirements for Serial Communication Functions

|  |  |
| --- | --- |
| Function Name | Sub-Module called by (Serial Comm Function Name) |
| None |  |

# 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 |
| PwrLmtFuncCr\_Init1 | RTE\_START\_SEC\_AP\_PWRLMTFUNCCR\_APPL\_CODE |
| PwrLmtFuncCr\_Per1 | RTE\_START\_SEC\_AP\_PWRLMTFUNCCR\_APPL\_CODE |
| PwrLmtFuncCr\_Per2 | RTE\_START\_SEC\_AP\_PWRLMTFUNCCR\_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-19B v000B) | 7-Aug-12 | OT |
| 2 | 2.0 | Added checkpoints and memmap software segment is updated for static variables | 23-Sep-12 | Selva |
| 3 | 3.0 | Updated to version 2 to FDD 19 B | 23-Jan-13 | Selva |
| 4 | 4.0 | Apply limit else in Power Limit function corrected | 28-Jan-13 | Selva |
| 5 | 5.0 | Created local copies to module level variables in per2 | 29-Jan-13 | Selva |
| 6 | 6 | Added Low Pass Filter to Motor Velocity | 04-Feb-13 | LN |
| 7 | 7.0 | Updated to FDD ver 004 (Fixes the anomaly 4686) | 13-Apr-13 | SP |
| 8 | 8.0 | Updated to FDD ver 006 | 21-May-13 | SP |
| 9 | 9.0 | Anomaly 5271 Fix, removed division from Power Limit function slew rate min/max values | 23-Jul-13 | VT |
| 10 | 10.0 | Update for FDD version 007 – Mapped Threshold\_Exceeded signal to NTC 0x0B2 (Reduced Assist due to Low Voltage) and update to TrqEnvLmt1\_MtrNm calculation. Added output limiting, divide by zero protection, and overflow protection. Also updated input, output, and module and display variable names per FDD and naming conventions. | 27-Aug-13 | KMC |
| 11 | 11.0 | Removed unnecessary divide by zero protection (that was added in version 10). Type casting update for QAC. | 11-Sep-13 | KMC |