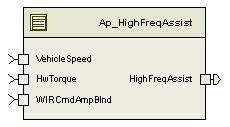
# Module –

# High-Level Description

This module compensates for system inertia and road feedback. It puts handwheel torque through a high-pass filter and multiplies it by a tunable gain parameter to compensate for these factors.

# 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 | |
| VehicleSpeed\_Kph\_f32 | | HighFreqAssist\_MtrNm\_f32 |
| HwTorque\_HwNm\_f32 | |  |
| WIRCmdAmpBlnd\_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 |
| HwTorqueHPFKSV\_Cnt\_M\_str | Single Precision Float | -10 | 10 | HYSTADD\_START\_SEC\_VAR\_CLEARED\_UNSPECIFIED |
| GainBlend\_Uls\_D\_f32 | Single Precision Float | 0 | 1 | HIGHFREQASSIST\_START\_SEC\_VAR\_CLEARED\_32 |
| GainWIRZero\_MtrNmpHwNm\_D\_f32 | Single Precision Float | 0 | 10 | HIGHFREQASSIST\_START\_SEC\_VAR\_CLEARED\_32 |
| GainVal\_MtrNmpHwNm\_D\_f32 | Single Precision Float | 0 | 10 | HIGHFREQASSIST\_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) |
| 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 |
| t\_LPFKnY\_Hz\_u7p9[12] |
| t\_CmnVehSpd\_Kph\_u9p7[12] |
| t2\_TorqX0\_HwNm\_u5p11[12][13] |
| t2\_TorqX1\_HwNm\_u5p11[12][13] |
| t2\_GainY0\_MtrNmpHwNm\_u3p13[12][13] |
| t2\_GainY1\_MtrNmpHwNm\_u3p13[12][13] |
| t2\_WIRBlendX\_MtrNm\_u4p12[12][5] |
| t2\_WIRBlendY\_Uls\_u1p15[12][5] |

## 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\_2MS\_SEC\_F32 |

### 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. FPM\_FloatToFixed\_m
2. FPM\_FixedToFloat\_m
3. Abs\_s16\_m
4. IntplVarXY\_u16\_u16Xu16Y\_Cnt
5. BilinearXMYM\_u16\_u16XMu16YM\_Cnt
6. TableSize\_m
7. Blend\_f32
8. HPF\_KUpdate\_f32\_m
9. HPF\_OpUpdate\_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\_HighFreqAssist\_MtrNm\_f32 | 0 |
| Rte\_InitValue\_HwTorque\_HwNm\_f32 | 0 |
| Rte\_InitValue\_VehicleSpeed\_Kph\_f32 | 0 |
| Rte\_InitValue\_WIRCmdAmpBlnd\_MtrNm\_f32 | 0 |

## Initialization Functions

None

## Periodic Functions

### Per: \_Per1

#### Design Rationale

None

#### Program Flow Start

Rte\_Call\_HighFreqAssist\_Per1\_CP0\_CheckpointReached()

#### Store Module Inputs to Local copies

HwTorque\_HwNm\_T\_f32 = Rte\_IRead\_HighFreqAssist\_Per1\_HwTorque\_HwNm\_f32()

VehicleSpeed\_Kph\_T\_f32 = Rte\_IRead\_HighFreqAssist\_Per1\_VehicleSpeed\_Kph\_f32()

WIRCmdAmpBlnd\_MtrNm\_T\_f32 = Rte\_IRead\_HighFreqAssist\_Per1\_WIRCmdAmpBlnd\_MtrNm\_f32()

HwTorque\_HwNm\_T\_s4p11 = FPM\_FloatToFixed\_m(HwTorque\_HwNm\_T\_f32, s4p11\_T)

AbsHwTorque\_HwNm\_T\_u5p11 = Abs\_s16\_m(HwTorque\_HwNm\_T\_s4p11)

VehicleSpeed\_Kph\_T\_u9p7 = FPM\_FloatToFixed\_m(VehicleSpeed\_Kph\_T\_f32, u9p7\_T)

WIRCmdAmpBlnd\_MtrNm\_T\_u4p12 = FPM\_FloatToFixed\_m(WIRCmdAmpBlnd\_MtrNm\_T\_f32, u4p12\_T)

#### Determine Filter Frequency



#### Determine Gain



#### Filter and Output

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

GainBlend\_Uls\_D\_f32 = GainBlend\_Uls\_T\_f32

GainWIRZero\_MtrNmpHwNm\_D\_f32 = GainWIRZero\_MtrNmpHwNm\_T\_f32

GainVal\_MtrNmpHwNm\_D\_f32 = GainVal\_MtrNmpHwNm\_T\_f32

Rte\_IWrite\_HighFreqAssist\_Per1\_HighFreqAssist\_MtrNm\_f32(HighFreqAssist\_MtrNm\_T\_f32);

#### Program Flow End

Rte\_Call\_HighFreqAssist\_Per1\_CP1\_CheckpointReached()

## Fault Recovery Functions

None

## Shutdown Functions

None

## Interrupt Functions

None

## Serial Communication Functions

None

# Execution Requirements

## Execution Sequence of the Module

The periodic function is called at a rate of 2ms in all states.

## 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 |
| HighFreqAssist\_Per1 | 2 ms | ALL |

## 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 |
| HighFreqAssist\_Per1 | RTE\_START\_SEC\_AP\_HIGHFREQASSIST\_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 in GlobalMacro.h are not unit tested

# Revision Control Log

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item #** | **Rev #** | **Change Description** | **Date** | **Author Initials** |
| 1 | 1.0 | Initial Version | 05-Apr-12 | OT |
| 2 | 2.0 | Check points added for the runnable executables | 21-Sep-12 | SSK |
| 3 | 3.0 | Updated to FDD ver 002 | 2-May-13 | Jared |