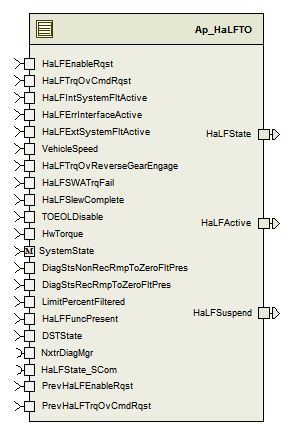
# Module -- HaLFTO

# High-Level Description

This function describes the activation logic used to implement the torque overlay functionality related to lane departure warning with haptic lane feedback.

# Figures

## Component Diagram



### Diagram – Function HaLFTO\_Per1

This diagram describes the functional characteristics and data flow of a given function.



# 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 | |
| HaLFEnableRqst\_Cnt\_lgc | | HaLFActive\_Cnt\_lgc |
|  | | HaLFState\_Cnt\_u08 |
| HaLFIntSystemFltActive\_Cnt\_lgc | | HaLFSuspend\_Cnt\_lgc |
| HaLFErrInterfaceActive\_Cnt\_lgc | |  |
| HaLFExtSystemFltActive\_Cnt\_lgc | |  |
| VehicleSpeed\_Kph\_f32 | |  |
| HaLFSWATrqFail\_Cnt\_lgc | |  |
| HaLFTrqOvReverseGearEngage\_Cnt\_lgc | |  |
| HaLFSlewComplete\_Cnt\_lgc | |  |
| HaLFFuncPresent\_Cnt\_lgc | |  |
| HwTorque\_HwNm\_f32 | |  |
| SystemState\_Mode | |  |
| LimitPercentFiltered\_Uls\_f32 | |  |
| DSTState\_Cnt\_u08 | |  |
| TOEOLDisable\_Cnt\_lgc | |  |
| PrevHaLFEnableRqst\_Cnt\_lgc | |  |
| PrevHaLFTrqOvCmdRqst\_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 |
| HaLFTO\_DeactDSTStateChkFail\_Cnt\_D\_lgc | 1 | FALSE | TRUE | HALFTO\_START\_SEC\_VAR\_CLEARED\_BOOLEAN |
| HaLFTO\_IncorHaLFActvnFailed\_Cnt\_M\_lgc | 1 | FALSE | TRUE | HALFTO\_START\_SEC\_VAR\_CLEARED\_BOOLEAN |
| HaLFTO\_HaLFDeactLongFailed\_Cnt\_M\_lgc | 1 | FALSE | TRUE | HALFTO\_START\_SEC\_VAR\_CLEARED\_BOOLEAN |
|  |  |  |  |  |
| HaLFTO\_IncorHaLFActvnHwTrqTimer\_mS\_M\_u32 | 1 | FULL | FULL | HALFTO\_START\_SEC\_VAR\_NOINIT\_32 |
| HaLFTO\_IncorHaLFActvnVehSpdTimer\_mS\_M\_u32 | 1 | FULL | FULL | HALFTO\_START\_SEC\_VAR\_NOINIT\_32 |
| HaLFTO\_IncorHaLFActvnRevGearTimer\_mS\_M\_u32 | 1 | FULL | FULL | HALFTO\_START\_SEC\_VAR\_NOINIT\_32 |
| HaLFTO\_HaLFDeactHwTrqTimer\_mS\_M\_u32 | 1 | FULL | FULL | HALFTO\_START\_SEC\_VAR\_NOINIT\_32 |
| HaLFTO\_HaLFDeactVehSpdTimer\_mS\_M\_u32 | 1 | FULL | FULL | HALFTO\_START\_SEC\_VAR\_NOINIT\_32 |
| HaLFTO\_HaLFDeactRevGearTimer\_mS\_M\_u32 | 1 | FULL | FULL | HALFTO\_START\_SEC\_VAR\_NOINIT\_32 |
| HaLFTO\_DSTSyncTimer\_mS\_M\_u32 | 1 | FULL | FULL | HALFTO\_START\_SEC\_VAR\_NOINIT\_32 |
| HaLFTO\_LimitPercentFilteredTimer\_mS\_M\_u32 | 1 | FULL | FULL | HALFTO\_START\_SEC\_VAR\_NOINIT\_32 |
|  |  |  |  |  |
|  |  |  |  |  |
| HaLFTO\_State\_Cnt\_M\_u08 | 1 | 0 | 3 | HALFTO\_START\_SEC\_VAR\_CLEARED\_8 |

# 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\_HaLFActvHwTrqTime\_mS\_u16 |
| k\_HaLFActvMaxHwTrq\_mS\_f32 |
| k\_HaLFActvVehSpdTime\_mS\_u16 |
| k\_HaLFActvMinVehSpd\_Kph\_f32 |
| k\_HaLFActvMaxVehSpd\_Kph\_f32 |
| k\_HaLFActvRevGearTime\_mS\_u16 |
| k\_HaLFDeactHwTrqTime\_mS\_u16 |
| k\_HaLFDeactMaxHwTrq\_HwNm\_f32 |
| k\_HaLFDeactVehSpdTime\_mS\_u16 |
| k\_HaLFDeactMinVehSpd\_Kph\_f32 |
| k\_HaLFDeactMaxVehSpd\_Kph\_f32 |
| k\_HaLFDeactRevGearTime\_mS\_u16 |
| k\_HaLFDSTSyncTime\_mS\_u16 |
| k\_TrqOverlayLimitPerc\_Uls\_f32 |
| k\_TrqOverlaySuspendTime\_mS\_u16 |

## 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\_HALFSTATEINACTIVE\_CNT\_U08 | 1 | Counts | 0 |
| D\_HALFSTATEACTIVE\_CNT\_U08 | 1 | Counts | 1 |
| D\_HALFSTATEIHIBITED\_CNT\_U08 | 1 | Counts | 2 |
| D\_HALFSTATERECOVERABLE\_CNT\_U08 | 1 | Counts | 3 |
| D\_DSTACTIVE1\_CNT\_U08 | 1 | Counts | 5 |
| D\_DSTACTIVE2\_CNT\_U08 | 1 | Counts | 6 |
| D\_DSTACTIVE3\_CNT\_U08 | 1 | Counts | 7 |
| D\_INCORHALFACTVNMASK\_CNT\_U08 | 1 | Counts | 1 |
| D\_HALFDEACTLONGMASK\_CNT\_U08 | 1 | Counts | 2 |

#### 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 |
| FLT\_EPSILON |

# 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. Rte\_Call\_SystemTime\_GetSystemTime\_mS\_u32
3. Rte\_Call\_SystemTime\_DtrmnElapsedTime\_mS\_u16
4. Rte\_Call\_NxtrDiagMgr\_GetNTCFailed
5. Rte\_Call\_NxtrDiagMgr\_SetNTCStatus
6. Rte\_Call\_HaLFState\_SCom\_Transition

## Data Hiding Functions

1. Rte\_Mode\_SystemState\_Mode

## Local Functions/Macros Used by this MDD only

### Common HwTrqVehSpdRevGear Check

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Function Name** | HwTrqVehSpdRevGearCheck | Type | Min | Max | UTP Tol. |
| **Arguments Passed** | HwTrqTime\_mS\_T\_u16 | Uint16 | 0 | FULL | 0 |
|  | HwTrqTimerPtr\_mS\_T\_u32 | Uint32\* | 0 | FULL | 0 |
|  | MaxHwTrq\_HwNm\_T\_f32 | float32 | 0.0 | 10.0 | 0.00390625 |
|  | VehSpdTime\_mS\_T\_u16 | Uint16 | 0 | FULL | 0 |
|  | VehSpdTimerPtr\_mS\_T\_u32 | Uint32\* | 0 | FULL | 0 |
|  | MinVehSpd\_Kph\_T\_f32 | float32 | 0 | 255 | .0078125 |
|  | MaxVehSpd\_Kph\_T\_f32 | float32 | 0 | 255 | .0078125 |
|  | RevGearChkTime\_mS\_T\_u16 | Uint16 | 0 | FULL | 0 |
|  | RevGearTimerPtr\_mS\_T\_u32 | Uint32\* | 0 | FULL | 0 |
| **Return Value** | HaLFDiagStatus\_Cnt\_T\_enum | Uint8 NxtrDiagMgrStatus | 0 | 3 | 0 |

#### Description

HaLFEnableRqst\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_HaLFEnableRqst\_Cnt\_lgc()

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

HaLFTrqOvReverseGearEngage\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_HaLFTrqOvReverseGearEngage\_Cnt\_lgc()

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

HaLFDiagStatus\_Cnt\_T\_enum = NTC\_STATUS\_PASSED

#### Hw Trq Check



#### Vehicle Speed Check



#### Reverse Gear Check



# 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 |
| DSTState\_Cnt\_u08 | 0 |
| DiagStsNonRecRmpToZeroFltPres\_Cnt\_lgc | FALSE |
| DiagStsRecRmpToZeroFltPres\_Cnt\_lgc | FALSE |
| HaLFActive\_Cnt\_lgc | FALSE |
| HaLFEnableRqst\_Cnt\_lgc | FALSE |
| HaLFErrInterfaceActive\_Cnt\_lgc | FALSE |
| HaLFExtSystemFltActive\_Cnt\_lgc | FALSE |
| HaLFFuncPresent\_Cnt\_lgc | FALSE |
| HaLFIntSystemFltActive\_Cnt\_lgc | FALSE |
| HaLFSWATrqFail\_Cnt\_lgc | FALSE |
| HaLFSlewComplete\_Cnt\_lgc | FALSE |
| HaLFState\_Cnt\_u08 | 0 |
| HaLFSuspend\_Cnt\_lgc | FALSE |
|  | 0 |
| HwTorque\_HwNm\_f32 | 0 |
| LimitPercentFiltered\_Uls\_f32 | 0 |
| HaLFTrqOvReverseGearEngage\_Cnt\_lgc | FALSE |
| VehicleSpeed\_Kph\_f32 | 0 |
| TOEOLDisable\_Cnt\_lgc | FALSE |
| PrevHaLFEnableRqst\_Cnt\_lgc |  |
| PrevTrqOvCmdRqst\_MtrNm\_f32 |  |

## Initialization Functions

### Init: HaLFTO\_Init1(void)

#### Design Rationale

The init function here sets all timers to some valid value, and informs SCom that the module is transitioning into the inactive state.

#### Initial State Transition

Rte\_Call\_HaLFState\_SCom\_Transition(D\_HALFSTATEINACTIVE\_CNT\_U08)

#### Module Internal

Rte\_Call\_SystemTime\_GetSystemTime\_mS\_u32(&Time\_mS\_u32)

HaLFTO\_IncorHaLFActvnHwTrqTimer\_mS\_M\_u32 = Time\_mS\_u32

HaLFTO\_IncorHaLFActvnVehSpdTimer\_mS\_M\_u32 = Time\_mS\_u32

HaLFTO\_IncorHaLFActvnRevGearTimer\_mS\_M\_u32 = Time\_mS\_u32

HaLFTO\_HaLFDeactHwTrqTimer\_mS\_M\_u32 = Time\_mS\_u32

HaLFTO\_HaLFDeactVehSpdTimer\_mS\_M\_u32 = Time\_mS\_u32

HaLFTO\_HaLFDeactRevGearTimer\_mS\_M\_u32 = Time\_mS\_u32

HaLFTO\_DSTSyncTimer\_mS\_M\_u32 = Time\_mS\_u32

HaLFTO\_LimitPercentFilteredTimer\_mS\_M\_u32 = Time\_mS\_u32

(void)Rte\_Call\_NxtrDiagMgr\_SetNTCStatus(NTC\_Num\_VLF\_04, 0x0U, NTC\_STATUS\_PASSED);

## Periodic Functions

### Per: \_Per1(void)

#### Design Rationale

None

#### Program Flow Start

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

#### Store Module Inputs to Local copies

DSTState\_Cnt\_T\_u08 = Rte\_IRead\_HaLFTO\_Per1\_DSTState\_Cnt\_u08()

DiagStsNonRecRmpToZeroFltPres\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_DiagStsNonRecRmpToZeroFltPres\_Cnt\_lgc()

DiagStsRecRmpToZeroFltPres\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_DiagStsRecRmpToZeroFltPres\_Cnt\_lgc()

HaLFEnableRqst\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_HaLFEnableRqst\_Cnt\_lgc()

HaLFErrInterfaceActive\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_HaLFErrInterfaceActive\_Cnt\_lgc()

HaLFExtSystemFltActive\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_HaLFExtSystemFltActive\_Cnt\_lgc()

HaLFFuncPresent\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_HaLFFuncPresent\_Cnt\_lgc()

HaLFIntSystemFltActive\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_HaLFIntSystemFltActive\_Cnt\_lgc()

HaLFSWATrqFail\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_HaLFSWATrqFail\_Cnt\_lgc()

HaLFSlewComplete\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_HaLFSlewComplete\_Cnt\_lgc()

LimitPercentFiltered\_Uls\_T\_f32 = Rte\_IRead\_HaLFTO\_Per1\_LimitPercentFiltered\_Uls\_f32()

TOEOLDisable\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_TOEOLDisable\_Cnt\_lgc()

PrevHaLFEnableRqst\_Cnt\_T\_lgc = Rte\_IRead\_HaLFTO\_Per1\_PrevHaLFEnableRqst\_Cnt\_lgc()

PrevTrqOvCmdRqst\_MtrNm\_T\_f32 = Rte\_IRead\_HaLFTO\_Per1\_PrevHaLFTrqOvCmdRqst\_MtrNm\_f32()

#### Initialize Faults



#### Incorrect HaLF Activation Diagnostic



#### HaLF Deactivation Diagnostic



#### HaLF Deactivation Diagnostic continued…



#### HalF Torque Overlay Enable



#### Transition Vector Logic



#### Inactive Transitions



#### Active Transitions



#### Recoverable Transitions



#### Transitions Complete



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

Rte\_IWrite\_HaLFTO\_Per1\_HaLFState\_Cnt\_u08(HaLFTO\_State\_Cnt\_M\_u08)

Rte\_IWrite\_HaLFTO\_Per1\_HaLFSuspend\_Cnt\_lgc(HaLFSuspend\_T\_lgc)

#### Program Flow End

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

## 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 |
| HaLFTO\_Per1 | 2ms | Warm Init, Disable, 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 |
| HaLFTO\_Init1 | RTE\_START\_SEC\_AP\_HALFTO\_APPL\_CODE |
| HaLFTO\_Per1 | RTE\_START\_SEC\_AP\_HALFTO\_APPL\_CODE |

## Local Functions

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

|  |  |
| --- | --- |
| Name of Sub Module | Software Segment |
| HwTrqVehSpdRevGearCheck | RTE\_AP\_HALFTO\_APPL\_CODE |

# Known Issues / Limitations With Design

1. Inline functions in GlobalMacro.h are not unit tested.

# Revision Control Log

|  |  |  |  |
| --- | --- | --- | --- |
| **Rev #** | **Change Description** | **Date** | **Author Initials** |
| 1 | Initial component creation. | 5-Nov-12 | BWL |
| 2 | Corrected vehicle speed check Incorrect HaLF activation(DST Active State Deactivation) | 20-Feb-13 | SR |
| 3 | Corrected Transition T5 from Recoverable to Inactive as per anomaly 4527 | 26-Feb-13 | SR |
| 4 | Update to FDD 40D v004 | 09-May-13 | BDO |
| 5 | Updated to CF 08A V001 | 09-Jul-13 | SP |
| 6 | Added logic to pass the NTCs if the enable criteria is FALSE. | 08-Oct-13 | MR |
| 7 | Updated to FDD CF-08C v004 | 23-Jan-14 | VT |
| 8 | Updated to FDD CF-08C v005 | 04-Feb-14 | VT |
| 9 | Unit Testing Finding Fixes | 20-Feb-14 | KPIT-PM |
| 10 | Updated to FDD CF-08C v006 | 24-Feb-14 | VT |
| 11 | Updated to FDD CF-08C v007 | 06-Mar-14 | VT |
| 12 | Updated per Design Review and Updated to FDD CF-08Cv008 | 24-Apr-14 | M. Story |
| 13 | A6806 anomaly fix 11959 | 20-May-14 | SB |