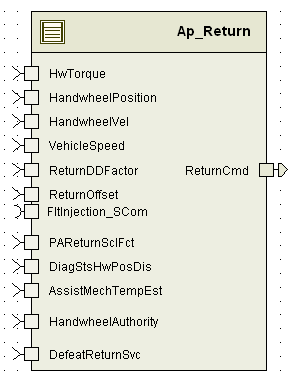
# Module -- Return

# High-Level Description

This function uses the Absolute Hand Wheel position, Hand Wheel Torque, Hand Wheel Velocity and Vehicle Speed to derive the desired Return Torque command.

# Figures

## Component Diagram



None

### Diagram – \_L5\_Per

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



# Module Inputs and Outputs

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

(Note: Full variable names required in table.)

(Note: All global variables including End Of Line data used should be shown here)

|  |  |
| --- | --- |
| Module Inputs (Global Variable Name) | Module Outputs (Global Variable Name) |
| HandwheelVel\_HwRadpS\_f32 | ReturnCmd\_MtrNm\_f32 |
| HandwheelAuthority\_Uls\_f32 |  |
| HandwheelPosition\_HwDeg\_f32 |  |
| HwTorque\_HwNm\_f32 |  |
| VehicleSpeed\_Kph\_f32 |  |
| SrlComSvcDft\_Cnt\_b32 |  |
| ReturnDDFactor\_Uls\_f32 |  |
| PAReturnSclFct\_Uls\_f32 |  |
| Return Offset\_HwDeg\_f32 |  |
| AssistMechTempEst\_DegC\_f32 |  |
| DefeatReturnSvc\_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.

(Note: If no module specific variables are used by the design, place the text “None” in the first Variable Name cell in the table)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Variable Name | Resolution | Legal Range  (min) | Legal Range  (max) | Software Segment |
| CurrentOffset\_HwDeg\_M\_f32 | Single Precision Float | 0 | 20 | RETURN\_START\_SEC\_VAR\_CLEARED\_32 |
| CrntHandWheelAthScl\_Uls\_M\_f32 | Single Precision Float | 0 | 1 | RETURN\_START\_SEC\_VAR\_CLEARED\_32 |
| HwPosReturnCmd \_MtrNm\_D\_f32 | Single Precision Float | 0 | 0.5 | RETURN\_START\_SEC\_VAR\_CLEARED\_32 |
| HwTrqReturnScl\_Uls \_D\_f32 | Single Precision Float | 0 | 1 | RETURN\_START\_SEC\_VAR\_CLEARED\_32 |
| HwVelReturnScl\_Uls \_D\_f32 | Single Precision Float | 0 | 50 | RETURN\_START\_SEC\_VAR\_CLEARED\_32 |
| TempReturnScl\_Uls\_D\_f32 | Single Precision Float | 0 | 10 | RETURN\_START\_SEC\_VAR\_CLEARED\_32 |
| AbsHwPosReturn\_HwDeg\_D\_u12p4 | 0.0625 | 0 | 1640 | RETURN\_START\_SEC\_VAR\_CLEARED\_16 |
| SgnHwPosReturn\_HwDeg\_D\_f32 | Single Precision Float | -1 | 1 | RETURN\_START\_SEC\_VAR\_CLEARED\_32 |
| RtrnBasicReturn\_MtrNm\_D\_f32 | SinglePrecisionFloating point | -10 | 10 | RETURN\_START\_SEC\_VAR\_CLEARED\_32 |

### User defined typedef definition/declaration

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

|  |  |  |  |
| --- | --- | --- | --- |
| Variable Name | Typedef Name | Storage Type | Safety Critical Classification |
| 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.

(Note: If no calibrations are used by the design, place the text “None” in the first location in the table)

|  |
| --- |
| Constant Name |
| t\_ReturnVSpdTblBS\_Kph\_u9p7 |
| t2\_ReturnPosTblXM\_HwDeg\_u12p4 |
| t2\_ReturnPosTblYM\_MtrNm\_u5p11 |
| t2\_ReturnSclTrqTblXM\_HwNm\_T\_u8p8 |
| t2\_ReturnSclTrqTblYM\_Uls\_u8p8 |
| t2\_ReturnSclVelTblXM\_HwRadpS\_T\_u7p9 |
| t2\_ReturnSclVelTblYM\_Uls\_u8p8 |
| k\_RtnOffsetSlew\_HwDegpS\_f32 |
| k\_RtnOffsetRange\_HWDeg\_f32 |
| t\_ReturnTempScaleXTbl\_DegC\_s11p4 |
| t\_ReturnTempScaleSclYTbl\_Uls\_u8p8 |
| t\_HWAuthRetScl\_X\_Uls\_u8p8 |
| t\_HWAuth RetScl\_Y\_Uls\_u9p7 |
| k\_RtnHWAuthSlew\_UlspS\_f32 |
| k\_RtnLimit\_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 | Value |
|  |  |  |
|  |  |  |

#### Note: RtnLoopTime depends on the rate of the periodic function.

#### 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 |
| BC\_RETURN\_FAULTINJECTIONPOINT |
| STD\_ON |
| FLTINJ\_RETURN |
| D\_2MS\_SEC\_F32 |

### Module specific Lookup Tables Constants

(This is for lookup tables (arrays) with fixed values, same name as other tables)

|  |  |  |  |
| --- | --- | --- | --- |
| Constant Name | Resolution | Value | Software Segment |
|  |  |  |  |

# Software Module Implementation

## Initialization Functions

### Init: \_L5\_Init()

#### Design Rationale

None

#### Module Outputs

None

#### Module Internal

None

## Periodic Functions

### Per: \_L5\_Per

#### Design Rationale

None

#### Program Flow Start

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

#### Store Module Inputs to Local copies

HwVel\_HwRadpS\_T\_f32 = Rte\_IRead\_Return\_Per1\_HandwheelVel\_HwRadpS\_f32 ()

FinHwPosAuth\_Uls\_T\_f32 = Rte\_IRead\_Return\_Per1\_HandwheelAuthority\_Uls\_f32 ()

FinHwPos\_HwDeg\_T\_f32 = Rte\_IRead\_Return\_Per1\_HandwheelPosition\_HwDeg\_f32 ()

HwTrq\_HwNm\_T\_f32 = Rte\_IRead\_Return\_Per1\_HwTorque\_HwNm\_f32 ()

DrvDynSclFct\_Uls\_T\_f32 = Rte\_IRead\_Return\_Per1\_ReturnDDFactor\_Uls\_f32()

PrkAstSclFct\_Uls\_T\_f32 = Rte\_IRead\_Return\_Per1\_PAReturnSclFct\_Uls\_f32 ()

AssistMechTempEst\_T\_DegC\_f32 = Rte\_IRead\_Return\_Per1\_AssistMechTempEst\_DegC\_f32()

ReturnOffset\_HwDeg\_T\_f32 = Rte\_IRead\_Return\_Per1\_ReturnOffset\_HwDeg\_f32()

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

DiagStsHwPosDis\_Cnt\_T\_lgc = Rte\_IRead\_Return\_Per1\_ DiagStsHwPosDis\_Cnt\_lgc ()

DefeatReturnSvc\_Cnt\_T\_lgc = Rte\_IRead\_Return\_Per1\_DefeatReturnSvc\_Cnt\_lgc()

VehSpd\_Kph\_T\_u9p7 = FPM\_FloatToFixed\_m(VehSpd\_Kph\_T\_f32, u9p7\_T)FinHwPosAuth\_Uls\_T\_u8p8 = FPM\_FloatToFixed\_m(FinHwPosAuth\_Uls\_T\_f32, u8p8\_T)

FinHwPos\_HwDeg\_T\_s11p4 = FPM\_FloatToFixed\_m(FinHwPos\_HwDeg\_T\_f32, s11p4\_T)

HwTrq\_HwNm\_T\_s7p8 = FPM\_FloatToFixed\_m(HwTrq\_HwNm\_T\_f32, s7p8\_T)

HwVel\_HwRadpS\_T\_s6p9 = FPM\_FloatToFixed\_m(HwVel\_HwRadpS\_T\_f32, s6p9\_T)

AssistMechTempEst\_DegC\_T\_s11p4 = FPM\_FloatToFixed\_m(AssistMechTempEst\_DegC\_f32, s11p4\_T)

Local Variables:

HwVel\_HwRadpS\_T\_f32

FinHwPosAuth\_Uls\_T\_f32

FinHwPos\_HwDeg\_T\_f32

HwTrq\_HwNm\_T\_f32

VehSpd\_Kph\_T\_f32

VehSpd\_Kph\_T\_u9p7

FinHwPos\_HwDeg\_T\_s11p4

HwTrq\_HwNm\_T\_s7p8

HwVel\_HwRadpS\_T\_s6p9

HwPosReturnCmd\_MtrNm\_T\_u5p11

HwPosReturnCmd\_MtrNm\_T\_f32

HwTrqReturnScl\_Uls\_T\_u8p8

HwTrqReturnScl\_Uls\_T\_f32

HwVelReturnScl\_Uls\_T\_u8p8

HwVelReturnScl\_Uls\_T\_f32

ReturnCmd\_MtrNm\_T\_f32

DrvDynSclFct\_Uls\_T\_f32

PrkAstSclFct\_Uls\_T\_f32

AssistMechTempEst\_T\_DegC\_f32

ReturnOffset\_HwDeg\_T\_f32

FinHwPosAuth\_Uls\_T\_u8p8

AssistMechTempEst\_DegC\_T\_s11p4

TempReturnScl\_Uls\_T\_u8p8

TempReturnScl\_Uls\_f32

ScaledReturn\_MtrNm\_f32

HandWheelAthScl\_Pct\_T\_u10p6

CrntHandWheelAthScl\_Uls\_f32

OffsetDiff\_HwDeg\_T\_u9p7

OffsetDiff\_HwDeg\_T\_u10p6

HandWheelAthScl\_Uls\_T\_u10p6

DiagStsHwPosDis\_Cnt\_T\_lgc

EOLRtnRange\_MtrNm\_T\_f32

HwPosReturnCmd\_MtrNm\_T\_f32

CurrentOffset\_HwDeg\_T\_s11p4

HandWheelAthScl\_Uls\_T\_f32

DefeatReturnSvc\_Cnt\_T\_lgc

#### Determine Return Command

##### Input Conditioning

##### 

##### Hand Wheel Position Return Command

The return command is determined as a function of Hand Wheel Position and Vehicle Speed.



##### Hand Wheel Torque Return Multiplier

The return command is scaled between 0 and 100 percent as a function of Hand Wheel Torque and Vehicle Speed.



##### Hand Wheel Velocity Return Multiplier

The return command is scaled between 0 and 100 percent as a function of Hand Wheel Velocity and Vehicle Speed.



##### Calculate Temperature Dependant Return Multiplier



##### Return Scale



##### Calculate Return Torque





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

Rte\_IWrite\_Return\_Per1\_ReturnCmd\_MtrNm\_f32(ReturnCmd\_MtrNm\_T\_f32)

HwPosReturnCmd\_MtrNm\_D\_f32 = HwPosReturnCmd\_MtrNm\_T\_f32

HwTrqReturnScl\_Uls\_D\_f32 = HwTrqReturnScl\_Uls\_T\_f32

HwVelReturnScl\_Uls\_D\_f32 = HwVelReturnScl\_Uls\_T\_f32

TempReturnScl\_Uls\_D\_f32 = TempReturnScl\_Uls\_f32

#### Program Flow End

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

## Fault Recovery Functions

None

## Shutdown Functions

None

## Interrupt Functions

None

## Serial Communication Functions

None

## Local Function/Macro Definitions

None

# Execution Requirements

## Execution Sequence of the Module

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

This table serves as reference for the Scheduler design

|  |  |  |  |
| --- | --- | --- | --- |
| Function Name | Task List | Calling Frequency | System State(s) in which the function is called |
| \_L5\_Per() |  | 2ms | 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 |
| \_L5\_Per() | RTE\_AP\_RETURN\_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. None

# Revision Control Log

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item #** | **Rev #** | **Change Description** | **Date** | **Author Initials** |
| 1 | 1.0 | Initial release | 16MAY11 | SAH |
| 2 | 2 | Apply scale factors from driving-dynamics and parking-assist. Replace computed handwheel velocity with global input. | 01-Jun-11 | YY |
| 3 | 3 | Updated for FDD #SF-02 | 17\_Nov\_11 | M. Story |
| 4 | 4 | Fixed Macro Names | 29 Nov11 | M. Story |
| 5 | 5 | Version 5 did not get checked in properly – so changes were lost and are in version 6 | 03Dec2011 | SMW |
| 6 | 6 | Updated calibration sign for temperature (need to be signed) and also update global input reads to match latest component template | 05Dec11 | SMW |
| 8 | 8 | Updated Globals based on UTP issue | 07Dec11 | M. Story |
| 9 | 9 | Corrected display variable assignments | 21Dec11 | M. Story |
| 10 | 10 | Changed ReturnRange to float32 from Uint16 | 04PR12 | M. Story |
| 11 | 11 | Added FLTINJ and some updates as per SF02 Ver004 | 10May12 | NRAR |
| 12 | 12 | Anom 3317 fix for RateLimiter scaling issue | 16 May 12 | NRAR |
| 13 | 13 | Move EOLRtnRange\_MtrNm\_f32 from EEPROM to flash, adding k\_RtnLimit\_MtrNm\_f32CR5873 | 20July12 | BDO |
| 14 | 14 | Added checkpoints and memmap software segment is updated for static variables | 23-Sep-2012 | Selva |
| 15 | 15 | Corrected anomaly 4893 | 27-Apr-13 | KJS |
| 16 | 16 | Updated to SF-02 V 006 | 23-May-13 | SP |