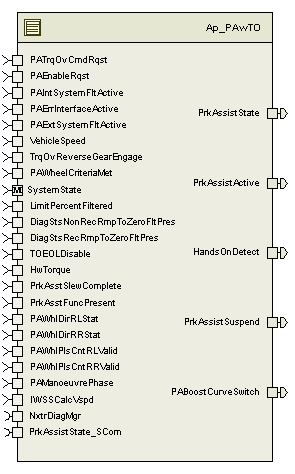
# Module --

# High-Level Description

This function describes the algorithms used to implement the activation logic for torque overlay functionality related to park assist.

# Figures

## Diagram – 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.

(Note: Full variable names required in table.)

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

|  |  |  |
| --- | --- | --- |
| Module Inputs | Module Outputs | |
| PATrqOvCmdRqst\_HwNm\_f32 | | PrkAssistActive\_Cnt\_lgc |
| PAEnableRqst\_Cnt\_lgc | | PrkAssistState\_Cnt\_u08 |
| PAIntSystemFltActive\_Cnt\_lgc | | HandsOnDetect\_Cnt\_lgc |
| PAErrInterfaceActive\_Cnt\_lgc | | PrkAssistSuspend\_Cnt\_lgc |
| PAExtSystemFltActive\_Cnt\_lgc | | PABoostCurveSwitch\_Cnt\_lgc |
| VehicleSpeed\_Kph\_f32 | |  |
| TrqOvReverseGearEngage\_Cnt\_lgc | |  |
| PAWheelCriteriaMet\_Cnt\_lgc | |  |
| LimitPercentFiltered\_Uls\_f32 | |  |
| DiagStsNonRecRmpToZeroFltPres\_Cnt\_lgc | |  |
| DiagStsRecRmpToZeroFltPres\_Cnt\_lgc | |  |
| TOEOLDisable\_Cnt\_lgc | |  |
| HwTorque\_HwNm\_f32 | |  |
| PrkAsstSlewComplete\_Cnt\_lgc | |  |
| PrkAsstFuncPresent\_Cnt\_lgc | |  |
| PAWhlDirRLStat\_Cnt\_u08 | |  |
| PAWhlDirRRStat\_Cnt\_u08 | |  |
| PAWhlPlsCntRLValid\_Cnt\_lgc | |  |
| PAWhlPlsCntRRValid\_Cnt\_lgc | |  |
| PAManoeuvrePhase\_Cnt\_u08 | |  |
| IWSSCalcVspd\_Kph\_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 |
| PAwTO\_PrkAssistState\_Cnt\_M\_u08 | 1 | 0 | 3 | PAWTO\_START\_SEC\_VAR\_CLEARED\_8 |
| PAwTO\_VSpdActvnMet\_Cnt\_M\_lgc | 1 | FALSE | TRUE | PAWTO\_START\_SEC\_VAR\_CLEARED\_BOOLEAN |
| PAwTO\_IWSSComputedSpin\_Cnt\_M\_u08 | 1 | 0 | 3 | PAWTO\_START\_SEC\_VAR\_CLEARED\_8 |
| PAwTO\_HandsOnHwTrqSV\_HwNm\_M\_s4p27 | 7.45058E-09 | -10 | 10 | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |
| PAwTO\_BoostCurveSwitchTimer\_mS\_M\_u32 | 1 | FULL | FULL | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |
| PAwTO\_LimitPercentFilteredTimer\_mS\_M\_u32 | 1 | FULL | FULL | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |
| PAwTO\_RvsGearTimer\_mS\_M\_u32 | 1 | FULL | FULL | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |
| PAwTO\_TrqOvRqNotZeroTimer\_mS\_M\_u32 | 1 | FULL | FULL | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |
| PAwTO\_VehSpdNotLowTimer\_mS\_M\_u32 | 1 | FULL | FULL | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |
| PAwTO\_IWSSWhlSpinTimer\_mS\_M\_u32 | 1 | FULL | FULL | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |
| PAwTO\_IWSSWhlSpinClrTimer\_mS\_M\_u32 | 1 | FULL | FULL | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |
| PAwTO\_ExcessVehSpdTimer\_mS\_M\_u32 | 1 | FULL | FULL | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |
| PAwTO\_ExcessVehSpdClrTimer\_mS\_M\_u32 | 1 | FULL | FULL | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |
| PAwTO\_MovMismatchTimer\_mS\_M\_u32 | 1 | FULL | FULL | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |
| PAwTO\_MovMismatchClrTimer\_mS\_M\_u32 | 1 | FULL | FULL | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |
| PAwTO\_VehicleSpeedMismatchTimer\_mS\_M\_u32 | 1 | FULL | FULL | PAWTO\_START\_SEC\_VAR\_NOINIT\_32 |

# 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\_HandsOnLPFKn\_Cnt\_u16 |
| k\_TrqOverlayHandsOnTrq\_HwNm\_u5p27 |
| k\_PARvsGearChkTime\_mS\_u16 |
| k\_PATrqOvNotZeroChkTime\_mS\_u16 |
| k\_PAVehSpdNotLowTime\_mS\_u16 |
| k\_PAVSpdNotLowMin\_Kph\_f32 |
| k\_PAVSpdNotLowMax\_Kph\_f32 |
| k\_PAIWSSWhlSpinSetTime\_mS\_u16 |
| k\_PAIWSSWhlSpinClrTime\_mS\_u16 |
| k\_PAExcessVehSpdSetTime\_mS\_u16 |
| k\_PAExcessVehSpdClrTime\_mS\_u16 |
| k\_PAExcessVehSpd\_Kph\_f32 |
| k\_PAMovMismatchSetTime\_mS\_u16 |
| k\_PAMovMismatchClrTime\_mS\_u16 |
| k\_PAMovMismatchVSpd\_Kph\_f32 |
| k\_PAVspdMismatchSetTime\_mS\_u16 |
| k\_PAEIWSSChkVspd\_Kph\_f32 |
| k\_PABoostCurveTime\_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\_PASTATEINACTIVE\_CNT\_U08 | 1 | Counts | 0 |
| D\_PASTATEACTIVE\_CNT\_U08 | 1 | Counts | 1 |
| D\_PASTATEINHIBITED\_CNT\_U08 | 1 | Counts | 2 |
| D\_PASTATERECOVERABLE\_CNT\_U08 | 1 | Counts | 3 |
| D\_STDSTIL\_CNT\_U08 | 1 | Counts | 0 |
| D\_FORWARD\_CNT\_U08 | 1 | Counts | 1 |
| D\_BACKWRD\_CNT\_U08 | 1 | Counts | 2 |
| D\_INVALID\_CNT\_U08 | 1 | Counts | 3 |
| D\_MANOEUVREPAHSEBKWD\_CNT\_U08 | 1 | Counts | 0 |
| D\_MANOEUVREPAHSEFWD\_CNT\_U08 | 1 | Counts | 1 |
| D\_PARVSGEARMASK\_CNT\_U08 | 1 | Counts | 1 |
| D\_PATRQOVNOTZEROMASK\_CNT\_U08 | 1 | Counts | 2 |
| D\_PAVSPDNOTLOW\_CNT\_U08 | 1 | Counts | 4 |

#### 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\_ONE\_CNT\_U8 |
| FLT\_EPSILON |

### 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 |
| T2\_IWSSCOMPSPIN\_CNT\_U08 | 1 | {  {D\_STDSTIL\_CNT\_U08, D\_FORWARD\_CNT\_U08, D\_BACKWRD\_CNT\_U08, D\_STDSTIL\_CNT\_U08},  {D\_FORWARD\_CNT\_U08, D\_FORWARD\_CNT\_U08, D\_FORWARD\_CNT\_U08, D\_FORWARD\_CNT\_U08},  {D\_BACKWRD\_CNT\_U08, D\_FORWARD\_CNT\_U08, D\_BACKWRD\_CNT\_U08, D\_BACKWRD\_CNT\_U08},  {D\_STDSTIL\_CNT\_U08, D\_FORWARD\_CNT\_U08, D\_BACKWRD\_CNT\_U08, D\_INVALID\_CNT\_U08}  }; | PAWTO\_START\_SEC\_CONST\_8 |

# 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. Rte\_Call\_NxtrDiagMgr\_SetNTCStatus
3. Rte\_Call\_NxtrDiagMgr\_GetNTCActive
4. Rte\_Call\_SystemTime\_GetSystemTime\_mS\_u32
5. Rte\_Call\_SystemTime\_DtrmnElapsedTime\_mS\_u16
6. LPF\_SvUpdate\_s16InFixKTrunc\_m
7. Abs\_s32\_m

## Data Hiding Functions

1. Rte\_Mode\_SystemState\_Mode

# 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 |
| DiagStsNonRecRmpToZeroFltPres\_Cnt\_lgc | FALSE |
| DiagStsRecRmpToZeroFltPres\_Cnt\_lgc | FALSE |
| HandsOnDetect\_Cnt\_lgc | FALSE |
| HwTorque\_HwNm\_f32 | 0 |
| IWSSCalcVspd\_Kph\_f32 | 0 |
| LimitPercentFiltered\_Uls\_f32 | 0 |
| PABoostCurveSwitch\_Cnt\_lgc | FALSE |
| PAEnableRqst\_Cnt\_lgc | FALSE |
| PAErrInterfaceActive\_Cnt\_lgc | FALSE |
| PAExtSystemFltActive\_Cnt\_lgc | FALSE |
| PAIntSystemFltActive\_Cnt\_lgc | FALSE |
| PAManoeuvrePhase\_Cnt\_u08 | 0 |
| PATrqOvCmdRqst\_HwNm\_f32 | 0 |
| PAWheelCriteriaMet\_Cnt\_lgc | FALSE |
| PAWhlDirRLStat\_Cnt\_u08 | 3 |
| PAWhlDirRRStat\_Cnt\_u08 | 3 |
| PAWhlPlsCntRLValid\_Cnt\_lgc | FALSE |
| PAWhlPlsCntRRValid\_Cnt\_lgc | FALSE |
| PrkAssistActive\_Cnt\_lgc | FALSE |
| PrkAssistState\_Cnt\_u08 | 0 |
| PrkAssistSuspend\_Cnt\_lgc | FALSE |
| PrkAsstFuncPresent\_Cnt\_lgc | FALSE |
| PrkAsstSlewComplete\_Cnt\_lgc | FALSE |
| TOEOLDisable\_Cnt\_lgc | FALSE |
| TrqOvReverseGearEngage\_Cnt\_lgc | FALSE |
| VehicleSpeed\_Kph\_f32 | 0 |

## Initialization Functions

### Init: PAwTO\_Init1(void)

#### Design Rationale

This init function initializes the hand wheel torque low pass filter, and the fault timers. Also initializes the state from the perspective of the SCom module.

#### Module Outputs

None

#### Module Internal

HwTorque\_HwNm\_T\_f32 = Rte\_IRead\_PAwTO\_Init1\_HwTorque\_HwNm\_f32()

PAwTO\_HandsOnHwTrqSV\_HwNm\_M\_s4p27 = FPM\_FloatToFixed\_m(HwTorque\_HwNm\_T\_f32, s4p27\_T)

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

PAwTO\_BoostCurveSwitchTimer\_mS\_M\_u32 = Time\_mS\_T\_u32

PAwTO\_LimitPercentFilteredTimer\_mS\_M\_u32 = Time\_mS\_T\_u32

PAwTO\_RvsGearTimer\_mS\_M\_u32 = Time\_mS\_T\_u32

PAwTO\_TrqOvRqNotZeroTimer\_mS\_M\_u32 = Time\_mS\_T\_u32

PAwTO\_VehSpdNotLowTimer\_mS\_M\_u32 = Time\_mS\_T\_u32

PAwTO\_IWSSWhlSpinTimer\_mS\_M\_u32 = Time\_mS\_T\_u32

PAwTO\_IWSSWhlSpinClrTimer\_mS\_M\_u32 = Time\_mS\_T\_u32

PAwTO\_ExcessVehSpdTimer\_mS\_M\_u32 = Time\_mS\_T\_u32

PAwTO\_ExcessVehSpdclrTimer\_mS\_M\_u32 = Time\_mS\_T\_u32

PAwTO\_MovMismatchTimer\_mS\_M\_u32 = Time\_mS\_T\_u32

PAwTO\_MovMismatchClrTimer\_mS\_M\_u32 = Time\_mS\_T\_u32

PAwTO\_VehicleSpeedMismatchTimer\_mS\_M\_u32 = Time\_mS\_T\_u32

Rte\_Call\_PrkAssistState\_SCom\_Transition(D\_PASTATEINACTIVE\_CNT\_U08)

## Periodic Functions

(Note: For multiple periodic functions, insert new headers at the “Header 2” level – subset of “5.2 Periodic Functions” and follow the same sub-section design shown below)

### Per: PAwTO\_Per1(void)

#### Design Rationale

This periodic handles all portions of PAwTO with a 2ms trigger rate.

#### Program Flow Start

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

#### Store Module Inputs to Local copies

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

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

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

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

PAEnableRqst\_Cnt\_T\_lgc = Rte\_IRead\_PAwTO\_Per1\_PAEnableRqst\_Cnt\_lgc()

PAErrInterfaceActive\_Cnt\_T\_lgc = Rte\_IRead\_PAwTO\_Per1\_PAErrInterfaceActive\_Cnt\_lgc()

PAExtSystemFltActive\_Cnt\_T\_lgc = Rte\_IRead\_PAwTO\_Per1\_PAExtSystemFltActive\_Cnt\_lgc()

PAIntSystemFltActive\_Cnt\_T\_lgc = Rte\_IRead\_PAwTO\_Per1\_PAIntSystemFltActive\_Cnt\_lgc()

PAWheelCriteriaMet\_Cnt\_T\_lgc = Rte\_IRead\_PAwTO\_Per1\_PAWheelCriteriaMet\_Cnt\_lgc()

PrkAsstFuncPresent\_Cnt\_T\_lgc = Rte\_IRead\_PAwTO\_Per1\_PrkAsstFuncPresent\_Cnt\_lgc()

PrkAsstSlewComplete\_Cnt\_T\_lgc = Rte\_IRead\_PAwTO\_Per1\_PrkAsstSlewComplete\_Cnt\_lgc()

TrqOvReverseGearEngage\_Cnt\_T\_lgc = Rte\_IRead\_PAwTO\_Per1\_TrqOvReverseGearEngage\_Cnt\_lgc()

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

VSpdActvnMet\_Cnt\_T\_lgc = PAwTO\_VSpdActvnMet\_Cnt\_M\_lgc

Rte\_Call\_NxtrDiagMgr\_GetNTCActive(NTC\_Num\_ExVoltageLow, &NTCLowBattVtg\_Cnt\_T\_lgc)

Rte\_Call\_NxtrDiagMgr\_GetNTCActive(NTC\_Num\_VLF\_02, &ParkAsstActvnFailed\_Cnt\_T\_lgc)

Rte\_Call\_NxtrDiagMgr\_GetNTCActive(NTC\_Num\_VLF\_08, &NTCPAVspdMismatch\_Cnt\_T\_lgc)

Rte\_Call\_NxtrDiagMgr\_GetNTCActive(NTC\_Num\_VLF\_05, &NTCPAIWSSVspdValidity\_Cnt\_T\_lgc)

Rte\_Call\_NxtrDiagMgr\_GetNTCActive(NTC\_Num\_VLF\_06, &NTCPAExcessVspd\_Cnt\_T\_lgc)

Rte\_Call\_NxtrDiagMgr\_GetNTCActive(NTC\_Num\_VLF\_07, &NTCPARationalityChk\_Cnt\_T\_lgc)

Rte\_Call\_NxtrDiagMgr\_GetNTCActive(NTC\_Num\_InvalidMsg\_W, &NTCTireCircmInvalid\_Cnt\_T\_lgc)

#### Hands On Detection



#### Process for PrkAssist\_Activate Flag



#### Process PA\_BoostCurv\_Switch Output



#### Processing for PrkAssist\_Suspend Flag



#### PrkAssist\_State



#### Inactive Transitions



#### Active Transitions



#### Recoverable Transitions



#### Transition Complete



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

Rte\_IWrite\_PawTO\_Per1\_HandsOnDetect\_Cnt\_lgc(HandsOnDetect\_Cnt\_T\_lgc)

Rte\_Iwrite\_PawTO\_Per1\_PABoostCurveSwitch\_Cnt\_lgc(PABoostCurveSwitch\_Cnt\_T\_lgc)

Rte\_Iwrite\_PawTO\_Per1\_PrkAssistActive\_Cnt\_lgc(PrkAssistActive\_Cnt\_T\_lgc)

Rte\_Iwrite\_PawTO\_Per1\_PrkAssistState\_Cnt\_u08(PrkAssistState\_Cnt\_T\_u08)

Rte\_Iwrite\_PawTO\_Per1\_PrkAssistSuspend\_Cnt\_lgc(PrkAssistSuspend\_Cnt\_T\_lgc)

#### Program Flow End

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

### Per: PawTO\_Per2(void)

#### Design Rationale

This periodic handles all portions of PawTO with a 10ms trigger rate.

#### Program Flow Start

Rte\_Call\_PawTO\_Per2\_CP0\_CheckpointReached()

#### Store Module Inputs to Local copies

IWSSCalcVspd\_Kph\_T\_f32 = Rte\_Iread\_PawTO\_Per2\_IWSSCalcVspd\_Kph\_f32()

PAEnableRqst\_Cnt\_T\_lgc = Rte\_Iread\_PawTO\_Per2\_PAEnableRqst\_Cnt\_lgc()

PAManoeuvrePhase\_Cnt\_u08 = Rte\_Iread\_PawTO\_Per2\_PAManoeuvrePhase\_Cnt\_u08()

PATrqOvCmdRqst\_HwNm\_T\_f32 = Rte\_Iread\_PawTO\_Per2\_PATrqOvCmdRqst\_HwNm\_f32()

PAWhlDirRLStat\_Cnt\_T\_u08 = Rte\_Iread\_PawTO\_Per2\_PAWhlDirRLStat\_Cnt\_u08()

PAWhlDirRRStat\_Cnt\_T\_u08 = Rte\_Iread\_PawTO\_Per2\_PAWhlDirRRStat\_Cnt\_u08()

PAWhlPlsCntRLValid\_Cnt\_T\_lgc = Rte\_Iread\_PawTO\_Per2\_PAWhlPlsCntRLValid\_Cnt\_lgc()

PAWhlPlsCntRRValid\_Cnt\_T\_lgc = Rte\_Iread\_PawTO\_Per2\_PAWhlPlsCntRRValid\_Cnt\_lgc()

TrqOvReverseGearEngage\_Cnt\_T\_lgc = Rte\_Iread\_PawTO\_Per2\_TrqOvReverseGearEngage\_Cnt\_lgc()

VehicleSpeed\_Kph\_T\_f32 = Rte\_Iread\_PawTO\_Per2\_VehicleSpeed\_Kph\_f32()

PrkAssistState\_Cnt\_T\_u08 = PawTO\_PrkAssistState\_Cnt\_M\_u08

Rte\_Call\_NxtrDiagMgr\_GetNTCActive(NTC\_Num\_VLF\_05, &NTCPAIWSSVspdValidity\_Cnt\_T\_lgc)

Rte\_Call\_NxtrDiagMgr\_GetNTCActive(NTC\_Num\_VLF\_06, &NTCPAExcessVspd\_Cnt\_T\_lgc)

Rte\_Call\_NxtrDiagMgr\_GetNTCActive(NTC\_Num\_VLF\_07, &NTCPARationalityChk\_Cnt\_T\_lgc)

PAWhlDirRLStat\_Cnt\_T\_u08 = Min\_m(PAWhlDirRLStat\_Cnt\_T\_u08, D\_INVALID\_CNT\_U08)

PAWhlDirRRStat\_Cnt\_T\_u08 = Min\_m(PAWhlDirRRStat\_Cnt\_T\_u08, D\_INVALID\_CNT\_U08)

PrkAsstFuncPresent\_Cnt\_T\_lgc = Rte\_IRead\_PAwTO\_Per2\_PrkAsstFuncPresent\_Cnt\_lgc()

#### Incorrect Park Assist Activation Diagnostic



#### Torque Overlay Zero Check



#### Vehicle Speed Check



#### IWSS Signal Validity Check and Wheel Spin Signal Processing



#### IWSS Wheel Spin Inactive



#### Calculate IWSS Computed Spin



#### Excessive Vehicle Speed Condition



#### Excess Veh Speed Inactive



#### Actual Vehicle Longitudinal Movement and PPPA Computed Movement Mismatch Condition



#### Move Mismatch Inactive



#### ESC and IWSS Vehicle Speed Signal Mismatch Condition



#### Program Flow End

Rte\_Call\_PawTO\_Per2\_CP1\_CheckpointReached()

# Execution Requirements

## Execution Sequence of the Module

The order of execution of the two periodic is not important.

## 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 |
| PawTO\_Per1 | 2ms | Warm Init, Disable, Operate |
| PawTO\_Per2 | 10ms | 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 |
| PawTO\_Init1 | RTE\_START\_SEC\_AP\_HALFTO\_APPL\_CODE |
| PawTO\_Per1 | RTE\_START\_SEC\_AP\_HALFTO\_APPL\_CODE |
| PawTO\_Per2 | 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 |
| None |  |

# 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 | Anomaly 4434,4534 PawTO\_IWSSWhlSpinTimer\_mS\_M\_u32 used multiple times | 21-MAR-13 | MRS |
| 3 | Anomaly fixes 4767,4768,4777 | 12-APR-13 | Srikanth |
| 4 | Anomaly 4815, 4816 | 18-Apr-13 | Srikanth |
| 5 | Update to FDD 40B v004 | 09-May-13 | BDO |
| 6 | Update to FDD 40B v005 | 19-Jun-13 | SP |
| 7 | Update to CF-08A v001 | 09-Jul-13 | SP |
| 8 | Added logic to pass the NTCs if the enable criteria is FALSE | 08-Oct13 | MR |
| 9 | Changed GetNTCFailed to GetNTCActive macro. | 22-Oct-13 | MR |
| 10 | Updated to CF-08A Park Assist rev 004 | 05-Feb-14 | VT |