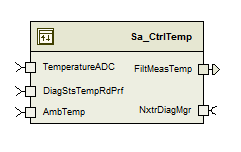
# Module -- Controller Temperature

# High-Level Description

This module monitors the controller’s temperature sensor output, filters that output, and checks whether the output is within a lower and upper limit.

# Figures



## Diagram – Function Data Sharing

This diagram shows all data that is shared between functions within the module.

# 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) |
| DiagStsTempRdPrf\_Cnt\_lgc | FiltMeasTemp\_DegC\_f32 |
| TemperatureADC\_Volt\_f32 |  |
| AmbTemp\_DegC\_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.

(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 |
| CtrlTemp\_CtrlTempSV\_M\_str | LPF32KSV\_Str | see data dictionary | see data dictionary | CTRLTEMP\_START\_SEC\_VAR\_CLEARED\_UNSPECIFIED |
| CtrlTemp\_CtrlTempSV\_M\_str .K\_Uls\_f32 | Single Precision Floating Point | see data dictionary | see data dictionary |  |
| CtrlTemp\_CtrlTempSV\_M\_str .SV\_Uls\_f32 | Single Precision Floating Point | see data dictionary | see data dictionary |  |
| CtrlTemp\_CtrlTemp\_DegC\_M\_f32 | Single Precision Floating Point | see data dictionary | see data dictionary | CTRLTEMP\_START\_SEC\_VAR\_CLEARED\_32 |
| CtrlTemp\_CtrlTempErrorAcc\_Cnt\_M\_u16 | 1 | see data dictionary | see data dictionary | CTRLTEMP\_START\_SEC\_VAR\_CLEARED\_16 |
| CtrlTemp\_CtrlTempFiltOut\_DegC\_D\_f32 | Single Precision Floating Point | see data dictionary | see data dictionary | CTRLTEMP\_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 |
| k\_TempSnsrFiltDft\_Cnt\_lgc |
| k\_TempSnsrLPFKn\_Hz\_f32 |
| k\_TempSnsrDefVal\_DegC\_f32 |
| k\_TempSensDiag\_Cnt\_str |
| k\_TempSensLowLimit\_DegC\_f32 |
| k\_TempSensHighLimit\_DegC\_f32 |
| k\_TempSnsrScaling\_DegpVolt\_f32 |
| k\_TempSnsrOffset\_Volts\_f32 |
| k\_CustDiagAmbLowThresh\_DegC\_f32 |
| k\_CustDiagAmbLowEnable\_Cnt\_lgc |

## 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 |
| D\_CTRLTEMPLOLMT\_DEGC\_F32 | Single Precision Floating Point | -50.0 |
| D\_CTRLTEMPHILMT\_DEGC\_F32 | Single Precision Floating Point | 150.0 |

#### 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 |
| D\_2MS\_SEC\_F32 |
| D\_ZERO\_ULS\_F32 |
| D\_FALSE\_CNT\_LGC |

### 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 |
|  |  |  |  |

# Functions/Macros used by the Sub-Modules

## Library Functions / Macros

The library functions / Macros that are called by the various sub modules are identified below,

1. LPF\_Init\_f32\_m
2. LPF\_OpUpdate\_f32\_m ()
3. Limit\_m()
4. DiagPStep\_m
5. DiagNStep\_m
6. DiagFailed\_m

## Data Hiding Functions

The data hiding functions / macros used in this module are identified below,

1. None

## Local Functions/Macros Used by this MDD only

1. None

# Software Module Implementation

## Initialization Functions

### Init: CtrlTemp\_ Init1

#### Design Rationale

None

#### Module Outputs

#### Module Internal

TempADC\_Volts\_T\_f32 = Rte\_IRead\_CtrlTemp\_Init1\_TemperatureADC\_Volt\_f32()

CtrlTemp\_DegC\_T\_f32 = (TempADC\_Volts\_T\_f32 – k\_TempSnsrOffset\_Volts\_f32) \* k\_TempSnsrScaling\_DegpVolt\_f32

CtrlTemp\_CtrlTemp\_DegC\_M\_f32 = Limit\_m(CtrlTemp\_DegC\_T\_f32, D\_CTRLTEMPLOLMT\_DEGC\_F32, D\_CTRLTEMPHILMT\_DEGC\_F32)

CtrlLPF\_Init\_f32\_m(CtrlTemp\_CtrlTemp\_DegC\_M\_f32, k\_TempSnsrLPFKn\_Hz\_f32, D\_2MS\_SEC\_F32, & CtrlTemp\_CtrlTempSV\_M\_str)

Rte\_IWrite\_CtrlTemp\_Init1\_FiltMeasTemp\_DegC\_f32 (CtrlTemp\_CtrlTemp\_DegC\_M\_f32)

## Periodic Functions

### Per: CtrlTemp\_Per1

#### Design Rationale

None

#### Program Flow Start

#### Rte\_Call\_CtrlTemp\_Per1\_CP0\_CheckpointReached()

#### Store Module Inputs to Local copies

DiagStsTempRdPrf\_Cnt\_T\_lgc = Rte\_Iread\_CtrlTemp\_Per1\_DiagStsTempRdPrf\_Cnt\_lgc();

TempADC\_Volts\_T\_f32 = Rte\_Iread\_CtrlTemp\_Per1\_TemperatureADC\_Volt\_f32()

CtrlTemp\_DegC\_T\_f32 = (TempADC\_Volts\_T\_f32 – k\_TempSnsrOffset\_Volts\_f32) \* k\_TempSnsrScaling\_DegpVolt\_f32

#### (Processing of function)………

##### Calculate Temperature

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

CtrlTemp\_DegC\_T\_f32 = **Limit\_m**(*CtrlTemp\_DegC\_T\_f32*, D\_CTRLTEMPLOLMT\_DEGC\_F32, D\_CTRLTEMPHILMT\_DEGC\_F32);

**Rte\_Iwrite\_CtrlTemp\_Per1\_FiltMeasTemp\_DegC\_f32** (CtrlTemp\_DegC\_T\_f32 );

#### Program Flow End

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

### Per: CtrlTemp\_Per2

#### Design Rationale

None

#### Program Flow Start

#### Rte\_Call\_CtrlTemp\_Per2\_CP0\_CheckpointReached()

#### Store Module Inputs to Local copies

#### (Processing of function)………



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

#### Program Flow End

Rte\_Call\_CtrlTemp\_Per2\_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 |
| CtrlTemp\_Init1() |  | Once | Once after RTE is started |
| CtrlTemp\_Per1() |  | 2ms | All |
| CtrlTemp\_Per2() |  | 100ms | 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 |
| CtrlTem\_Init1() | RTE\_SA\_CTRLTEMP\_APPL\_CODE |
| CtrlTem\_Per1() | RTE\_SA\_CTRLTEMP\_APPL\_CODE |
| CtrlTem\_Per2() | RTE\_SA\_CTRLTEMP\_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 | 18MAY11 | SAH |
| 2 | 2.0 | Added Reduced Performace temperature selecture per #SF-6 001 | 02DEC11 | M. Story |
| 3 | 3.0 | Anomaly 2995 Limits not set correctly | 29FEB12 | M. Story |
| 4 | 4.0 | Anomaly 2994 use CntrlTemp\_DegC\_T\_f32 temporary variable internal to the Per1 function | 03MAR12 | M. Story |
| 5 | 5.0 | Updated component to FDD SF-06 revision 4 | 16May12 | KJS |
| 6 | 6.0 | Updated floating point filter structure with K and SV ranges | 12Jun12 | KJS |
| 7 | 7.0 | Anomaly 3505 Use output from LPF for diagnostics. | 23Aug12 | Srikanth |
| 8 | 8.0 | Added watchdog checkpoints. | 16 Sept 12 | BWL |
| 9 | 9.0 | Added “Variables” missing from Module internal variable and their software segment | 18 sep 12 | SSK |
| 10 | 10.0 | Changed local constants to calibration to facilate the use of different temperature sensors. |  |  |
| 11 | 11.0 | Corrected naming conventions of new conversion calibrations. | 08Nov12 | LN |
| 12 | 12.0 | Corrected anomaly 4541 | 06Apr13 | KJS |
| 13 | 13.0 | Changed conditions on diagnostic per FDD SF—06 ver007; updated module and display variable names per naming conventions; updated temporary variable names for clarity; added missing section 5 and updated numbering on remaining sections | 17-Sep-13 | KMC |