**Module Design Document**

**For**

**DutyCycThermProtn**

**Oct 26, 2017**

**Prepared By:**

**TATA ELXSI,TRIVANDRUM, INDIA**

**Change History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Description** | **Author** | **Version** | **Date** |
| Initial Version | Sarika Natu(KPIT Technologies) | 1.0 | 02-Oct-2015 |
| Updated to version 2.0.0 of FDD | Krishna Anne | 2.0 | 07-Apr-2016 |
| Fix for anomaly EA4# 7558 | Krishna Anne | 3.0 | 29-Sep-2016 |
| Updated to FDD v3.0.0 | Shruthi Raghavan | 4.0 | 14-Dec-2016 |
| Updated as per FDD v4.0.0 | TATA | 5.0 | 25-Oct-2017 |

Table of Contents

[1 DutyCycThermProtn & High-Level Description 5](#_Toc469502658)

[2 Design details of software module 6](#_Toc469502659)

[2.1 Graphical representation of DutyCycThermProtn 6](#_Toc469502660)

[2.2 Data Flow Diagram 7](#_Toc469502661)

[2.2.1 Component level DFD 7](#_Toc469502662)

[2.2.2 Function level DFD 7](#_Toc469502663)

[3 Constant Data Dictionary 8](#_Toc469502664)

[3.1 Program (fixed) Constants 8](#_Toc469502665)

[3.1.1 Embedded Constants 8](#_Toc469502666)

[4 Software Component Implementation 9](#_Toc469502667)

[4.1 Sub-Module Functions 9](#_Toc469502668)

[4.1.1 Init: DutyCycThermProtn\_Init1 9](#_Toc469502669)

[4.1.1.1 Design Rationale 9](#_Toc469502670)

[4.1.1.2 Module Outputs 9](#_Toc469502671)

[4.1.2 Per: DutyCycThermProtn\_Per1 9](#_Toc469502672)

[4.1.2.1 Design Rationale 9](#_Toc469502673)

[4.1.2.2 Store Module Inputs to Local copies 9](#_Toc469502674)

[4.1.2.3 (Processing of function)……… 9](#_Toc469502675)

[4.1.2.4 Store Local copy of outputs into Module Outputs 9](#_Toc469502676)

[4.2 Server Runables 9](#_Toc469502677)

[4.3 Interrupt Functions 9](#_Toc469502678)

[4.4 Module Internal (Local) Functions 9](#_Toc469502679)

[4.4.1 Local Function #1 9](#_Toc469502680)

[4.4.1.1 Design Rationale 9](#_Toc469502681)

[4.4.1.2 Processing 10](#_Toc469502682)

[4.4.2 Local Function #2 10](#_Toc469502683)

[4.4.2.1 Design Rationale 10](#_Toc469502684)

[4.4.2.2 Processing 10](#_Toc469502685)

[4.4.3 Local Function #3 10](#_Toc469502686)

[4.4.3.1 Design Rationale 10](#_Toc469502687)

[4.4.3.2 Processing 10](#_Toc469502688)

[4.4.4 Local Function #4 10](#_Toc469502689)

[4.4.4.1 Design Rationale 11](#_Toc469502690)

[4.4.4.2 Processing 11](#_Toc469502691)

[4.4.5 Local Function #5 11](#_Toc469502692)

[4.4.5.1 Design Rationale 11](#_Toc469502693)

[4.4.5.2 Processing 11](#_Toc469502694)

[4.4.6 Local Function #6 11](#_Toc469502695)

[4.4.6.1 Design Rationale 11](#_Toc469502696)

[4.4.7 Local Function #7 11](#_Toc469502697)

[4.4.7.1 Design Rationale 12](#_Toc469502698)

[4.4.8 Local Function #8 12](#_Toc469502699)

[4.4.8.1 Design Rationale 12](#_Toc469502700)

[4.4.8.2 Processing 12](#_Toc469502701)

[4.5 GLOBAL Function/Macro Definitions 12](#_Toc469502702)

[5 Known Limitations with Design 13](#_Toc469502703)

[6 UNIT TEST CONSIDERATION 14](#_Toc469502704)

[Appendix A Abbreviations and Acronyms 15](#_Toc469502705)

[Appendix B Glossary 16](#_Toc469502706)

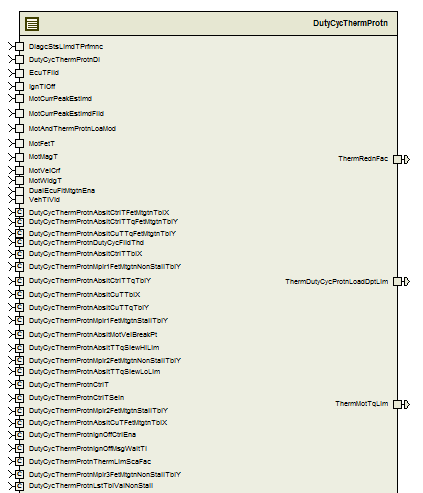
[Appendix C References 17](#_Toc469502707)

# DutyCycThermProtn & High-Level Description

The purpose of the Thermal Duty Cycle Protection is to limit and protect the system from excessive use, based on motor rotational velocity and system temperature. It also provides protection status information for use by other functions.

# Design details of software module

## Graphical representation of DutyCycThermProtn





## Data Flow Diagram

See FDD

### Component level DFD

See FDD

### Function level DFD

See FDD

# Constant Data Dictionary

## Program (fixed) Constants

### Embedded Constants

#### Local Constants

Refer .m file

|  |  |
| --- | --- |
| Constant Name | Value |
| THERMLOADLIMSIZE\_CNT\_U08 | 8U |
| MULTFILTERSIZE\_CNT\_U08 | 6U |
| BITMASK2\_CNT\_U08 | 2U |
| BITMASK4\_CNT\_U08 | 4U |
| IDX5\_CNT\_U08 | 5U |
| IDX8\_CNT\_U08 | 8U |

# Software Component Implementation

## Sub-Module Functions

## Init: DutyCycThermProtn\_Init1

## Design Rationale

*Refer FDD*

## Module Outputs

*Refer FDD*

## Per: DutyCycThermProtn\_Per1

## Design Rationale

DutyCycThermProtn\_Per1 function is divided into various functions to reduce the cyclomatic complexity.

The subsystems ‘Multiplier’ and ‘FilterPercMax’ are clubbed into ‘MultiFilterPercMax’ local function.

## Store Module Inputs to Local copies

*Refer FDD*

## (Processing of function)………

*Refer FDD*

## Store Local copy of outputs into Module Outputs

*Refer FDD*

## Server Runables

None

## Interrupt Functions

None

## Module Internal (Local) Functions

## Local Function #1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | FiltSVReinit | Type | Min | Max |
| **Arguments Passed** | IgnTiOff\_Cnt\_T\_u32 | uint32 | 0 | 1720000 |
|  | VehTiVld\_Cnt\_T\_Logl | Boolean | 0 | 1 |
| **Return Value** | None |  |  |  |

## Design Rationale

Name of local function matches with subsystem name from FDD

## Processing

## Local Function #2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | TemperatureSelection | Type | Min | Max |
| **Arguments Passed** | DiagcStsLimdTPrfmnc\_Cnt\_T\_Logl | boolean | 0 | 1 |
|  | EcuTFild\_DegCgrd\_T\_f32 | float32 | -50 | 150 |
|  | MotFetT\_DegCgrd\_T\_f32 | float32 | -50 | 200 |
|  | MotMagT\_DegCgrd\_T\_f32 | float32 | -50 | 150 |
|  | MotWidgT\_DegCgrd\_T\_f32 | float32 | -50 | 300 |
|  | \*Mult12Temp\_DegCgrd\_T\_ s15p0 | Sint16 | -50 | 200 |
|  | \*Mult36Temp\_DegCgrd\_T\_s15p0 | Sint16 | -50 | 300 |
| **Return Value** | SlcTemp\_DegCgrd\_T\_s15p0 | sint16 | -50 | 300 |

## Design Rationale

Name of local function matches with subsystem name from FDD

Note: The outputs of the function are Mult12Temp\_DegCgrd\_T\_s15p0, Mult36Temp\_DegCgrd\_T\_s15p0 and SlcTemp\_DegCgrd\_T\_f32.

## Processing

None

## Local Function #3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | TemperatureLimiting | Type | Min | Max |
| **Arguments Passed** | EcuTFild\_DegCgrd\_T\_f32 | float32 | -50 | 150 |
|  | MotWidgT\_DegCgrd\_T\_f32 | float32 | -50 | 300 |
| **Return Value** | AbsTempLimitSlew\_MotNwtMtr\_T\_f32 | float32 | 0 | 8.79 |

## Design Rationale

Name of local function matches with subsystem name from FDD

## Processing

None

## Local Function #4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | MultiFilterPercMax | Type | Min | Max |
| **Arguments Passed** | Mult12Temp\_DegCgrd\_T\_s15p0 | sint16 | -50 | 200 |
|  | Mult36Temp\_DegCgrd\_T\_s15p0 | sint16 | -50 | 300 |
|  | DutyCycThermProtnDi\_Cnt\_T\_Logl | boolean | 0 | 1 |
|  | MotVelCrf\_MotRadPerSec\_T\_f32 | float32 | -1350 | 1350 |
|  | MotCurrPeakEstimd\_AmprSqd\_T\_f32 | float32 | 0 | 62500 |
|  | MotCurrPeakEstimdFild\_AmprSqd\_T\_f32 | float32 | 0 | 62500 |
|  | \*MaxOut\_Uls\_T\_u16p0 | uint16 | 0 | 200 |
| **Return Value** | ThermLimSlowFilMax\_Uls\_T\_f32 | float32 | 0 | 200 |

## Design Rationale

The subsystems ‘Multiplier’ and ‘FilterPercMax’ are clubbed into ‘MultiFilterPercMax’ local function.

Note: The outputs of the function are MaxOut\_Uls\_T\_u16p0 and ThermLimSlowFilMax\_Uls\_T\_f32.

## Processing

None

## Local Function #5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ThermalLoadLimit | Type | Min | Max |
| **Arguments Passed** | MotVelCrf\_MotRadPerSec\_T\_f32 | float32 | -1350 | 1350 |
|  | SlcTemp\_DegCgrd\_T\_s15p0 | sint16 | -50 | 300 |
|  | MaxOut\_Uls\_T\_u16p0 | uint16 | 0 | 200 |
| **Return Value** | ThermalLoadLmt\_MotNwtMtr\_T\_f32 | float32 | 0 | 8.8 |

## Design Rationale

Name of local function matches with subsystem name from FDD

## Processing

None

## Local Function #6

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ThermalLimitStatus | Type | Min | Max |
| **Arguments Passed** | DutyCycThermProtnDi\_Cnt\_T\_Logl | Boolean | 0 | 1 |
|  | MaxOut\_Uls\_T\_u16p0 | uint16 | 0 | 200 |
|  | ThermMotTqLim\_MotNwtMtr\_T\_f32 | float32 | 0 | 8.8 |
| **Return Value** | ThermRednFac\_Uls\_T\_f32 | float32 | 0 | 1 |

## Design Rationale

Name of local function matches with subsystem name from FDD. Initializing ThermRednFac\_Uls\_T\_f32 to 0.0 helps to avoid writing another statement in the if-conditional (optimized compared to FDD)

## Local Function #7

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | TherrmalLimitScaling | Type | Min | Max |
| **Arguments Passed** | DualEcuFltMtgtnEna\_Cnt\_T\_logl | Boolean | 0 | 1 |
|  | IvtrLoaMtgtnEna\_Cnt\_T\_logl | Boolean | 0 | 1 |
|  | AbsTempLimitSlew\_MotNwtMtr\_T\_f32 | float32 | 0 | 8.79 |
|  | DutyCycThermProtnDi\_Cnt\_T\_Logl | Boolean | 0 | 1 |
|  | ThermalLoadLmt\_MotNwtMtr\_T\_f32 | float32 | 0 | 8.8 |
|  | \* ThermLoadDptLim\_MotNwtMtr\_T\_f32 | Float32 | 0 | 8.8 |
|  | \* ThermTempDptLim\_MotNwtMtr\_T\_f32 | Float32 | 0 | 8.8 |
| **Return Value** | ThermMotTqLim\_MotNwtMtr\_T\_f32 | float32 | 0 | 8.8 |

## Design Rationale

Name of local function matches with subsystem name from FDD

The if-action subsystem blocks for calculation of LoadDptLim and TempDptLim are clubbed together and optimized since the condition for the subsystem execution was same.

## Local Function #8

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | UseInpLowr | Type | Min | Max |
| **Arguments Passed** | \*TableX\_Cnt\_T\_s16 | sint16 | FULL | FULL |
|  | \*TableY\_Cnt\_T\_u16 | uint16 | FULL | FULL |
|  | Size\_Cnt\_T\_u16 | uint16 | 1 | 20 |
|  | Input\_Cnt\_T\_s16 | sint16 | FULL | FULL |
| **Return Value** | TableY\_Cnt\_T\_u16[Idx\_Cnt\_T\_u08] | uint16 | FULL | FULL |

## Design Rationale

None.

## Processing

None

## Local Function #9

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | Decoder | Type | Min | Max |
| **Arguments Passed** | MotAndThermProtnLoaMod\_Cnt\_T\_u08 | Uint8 | 0U | 255U |
| **Return Value** | IvtrLoaMtgtnEna\_Cnt\_T\_logl | boolean | FALSE | TRUE |

## Design Rationale

None.

## Processing

Refer to the “Decoder” block of the Simulink model of the design.

## GLOBAL Function/Macro Definitions

None

# Known Limitations with Design

1. **In Init function CurrMeasLoaMtgtnEna and FetLoaMtgtnEna are terminated. In Periodic1 CurrMeasLoaMtgtnEna is terminated. These flags need not be computed at all.**
2. **MotAndThermProtnLoaMod input readable to init runnable also. It is need to update in Data Dictionary.**

# UNIT TEST CONSIDERATION

* Function UseInpLowr to be tested only as called by the component; input and output ranges will not be reached.
* Function UseInpLowr’s TableX must have strictly increasing elements.

Abbreviations and Acronyms

| **Abbreviation or Acronym** | **Description** |
| --- | --- |
|  |  |
|  |  |

Glossary

**Note**: Terms and definitions from the source “Nexteer Automotive” take precedence over all other definitions of the same term. Terms and definitions from the source “Nexteer Automotive” are formulated from multiple sources, including the following:

* ISO 9000
* ISO/IEC 12207
* ISO/IEC 15504
* Automotive SPICE® Process Reference Model (PRM)
* Automotive SPICE® Process Assessment Model (PAM)
* ISO/IEC 15288
* ISO 26262
* IEEE Standards
* SWEBOK
* PMBOK
* Existing Nexteer Automotive documentation

| **Term** | **Definition** | **Source** |
| --- | --- | --- |
| MDD | Module Design Document |  |
| DFD | Data Flow Diagram |  |

References

| **Ref. #** | **Title** | **Version** |
| --- | --- | --- |
| 1 | AUTOSAR Specification of Memory Mapping (Link:[AUTOSAR\_SWS\_MemoryMapping.pdf](http://www.autosar.org/download/R4.0/AUTOSAR_SWS_MemoryMapping.pdf)) | v1.3.0 R4.0 Rev 2 |
| 2 | MDD Guideline | EA4 02.00.00 |
| 3 | [Software Naming Conventions.doc](http://misagweb01.nexteer.com/eRoomReq/Files/erooms8/NextGeneration/0_fc55f/Software%20Naming%20Conventions%2003x(In%20Work).doc) | 1.0 |
| 4 | [Software Design and Coding Standards.doc](http://eroom1.nexteer.com/eRoomReq/Files/erooms8/NextGeneration/0_1a67a9/Software%20Design%20and%20Coding%20Standards.doc) | 2.1 |
| 5 | FDD – SF009A\_DutyCycThermProtn\_Design | See Synergy sub project version |