**Module Design Document**

**For**

**‘PolarityCfg’**

**VERSION: 3.0**

**DATE: 07-Jul-2017**

**Prepared By:**

**Shawn Penning,**

**Nexteer Automotive,**

**Saginaw, MI, USA**

**Location:** The official version of this document is stored in the Nexteer Configuration Management System.

**Revision History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl. No.** | **Description** | **Author** | **Version** | **Date** |
| 1 | Initial Version | Sankardu Varadapureddi | 1.0 | 26-May-2015 |
| 3 | Corrections to Name and Graphic, Update MDD Layout | Shawn Penning | 3.0 | 07-Jul-2017 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Table of Contents**

[1 Polarity Configuration High-Level Description 6](#_Toc487189211)

[2 Design details of software module 7](#_Toc487189212)

[2.1 Graphical representation of Polarity Configuration 7](#_Toc487189213)

[2.2 Data Flow Diagram 7](#_Toc487189214)

[2.2.1 Module level DFD 7](#_Toc487189215)

[2.2.2 Sub-Module level DFD 7](#_Toc487189216)

[2.3 COMPONENT FLOW DIAGRAM 8](#_Toc487189217)

[3 Variable Data Dictionary 9](#_Toc487189218)

[3.1 User defined typedef definition/declaration 9](#_Toc487189219)

[3.2 Variable definition for enumerated types 9](#_Toc487189220)

[4 Constant Data Dictionary 10](#_Toc487189221)

[4.1 Program(fixed) Constants 10](#_Toc487189222)

[4.1.1 Embedded Constants 10](#_Toc487189223)

[4.1.1.1 Local 10](#_Toc487189224)

[4.1.1.2 Global 10](#_Toc487189225)

[4.1.2 Module specific Lookup Tables Constants 10](#_Toc487189226)

[5 Software Module Implementation 11](#_Toc487189227)

[5.1 Sub-Module Functions 11](#_Toc487189228)

[5.1.1 Initialization Functions 11](#_Toc487189229)

[5.1.1.1 INIT: PolarityCfgInit 11](#_Toc487189230)

[5.1.1.1.1 Design Rationale 11](#_Toc487189231)

[5.1.1.1.2 Module Outputs 11](#_Toc487189232)

[5.1.1.1.3 Module Internal 11](#_Toc487189233)

[5.1.2 PERIODIC FUNCTIONS 11](#_Toc487189234)

[5.1.3 Interrupt Functions 11](#_Toc487189235)

[5.1.4 Server runnables 12](#_Toc487189236)

[5.1.4.1 PolarityCfgRead 12](#_Toc487189237)

[5.1.4.1.1 Design Rationale 12](#_Toc487189238)

[5.1.4.1.2 Store Module Inputs to Local copies 12](#_Toc487189239)

[5.1.4.1.3 (Processing of function)……… 12](#_Toc487189240)

[5.1.4.1.4 Store Local copy of outputs into Module Outputs 12](#_Toc487189241)

[5.1.4.2 PolarityCfgWr 12](#_Toc487189242)

[5.1.4.2.1 Design Rationale 12](#_Toc487189243)

[5.1.4.2.2 Store Module Inputs to Local copies 12](#_Toc487189244)

[5.1.4.2.3 (Processing of function)……… 12](#_Toc487189245)

[5.1.4.2.4 Store Local copy of outputs into Module Outputs 12](#_Toc487189246)

[5.1.5 Local Function/Macro Definitions 12](#_Toc487189247)

[5.1.5.1 Local Function #1 12](#_Toc487189248)

[5.1.5.2 Description 12](#_Toc487189249)

[5.1.6 GLObAL Function/Macro Definitions 13](#_Toc487189250)

[5.1.7 Transition FUNCTIONS 13](#_Toc487189251)

[6 Known Limitations With Design 14](#_Toc487189252)

[7 UNIT TEST CONSIDERATION 15](#_Toc487189253)

[Appendix A Abbreviations and Acronyms 16](#_Toc487189254)

[Appendix B Glossary 17](#_Toc487189255)

[Appendix C References 18](#_Toc487189256)

# 



This section lists the title & version of all the documents that are referred for development of this document

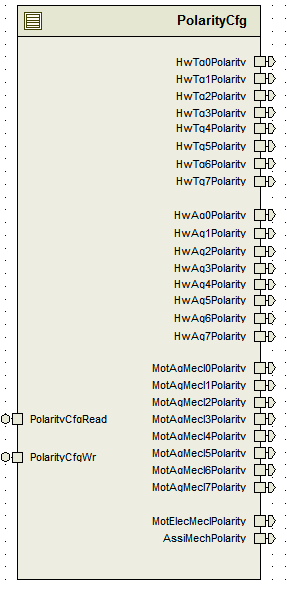
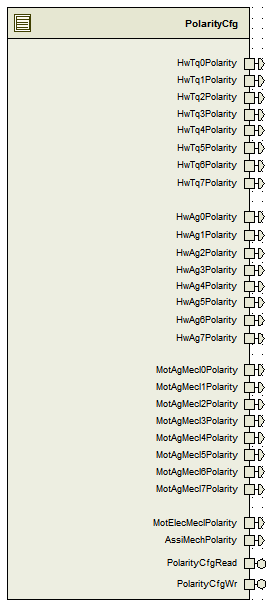


# Polarity Configuration High-Level Description

*This function will identify polarity control settings for certain points in the design.*

# Design details of software module

## Graphical representation of Polarity Configuration



## Data Flow Diagram

*Refer FDD*

## Module level DFD

*Refer FDD*

## Sub-Module level DFD

*Refer FDD*

## COMPONENT FLOW DIAGRAM

*Refer FDD*

# Variable Data Dictionary

## User defined typedef definition/declaration

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Typedef Name | Element Name | User Defined Type | Legal Range  (min) | Legal Range  (max) |
| None |  |  |  |  |
|  |  |  |  |  |

## Variable definition for enumerated types

|  |  |  |
| --- | --- | --- |
| Enum Name | Element Name | Value |
| None |  |  |

# Constant Data Dictionary

## Program(fixed) Constants

## Embedded Constants

## Local

|  |  |  |  |
| --- | --- | --- | --- |
| Constant Name | Resolution | Units | Value |
| HWAG0POL\_CNT\_U32 | Bitfield Mask | NA | 0x00000001U |
| HWAG1POL\_CNT\_U32 | Bitfield Mask | NA | 0x00000002U |
| HWAG2POL\_CNT\_U32 | Bitfield Mask | NA | 0x00000004U |
| HWAG3POL\_CNT\_U32 | Bitfield Mask | NA | 0x00000008U |
| HWAG4POL\_CNT\_U32 | Bitfield Mask | NA | 0x00000010U |
| HWAG5POL\_CNT\_U32 | Bitfield Mask | NA | 0x00000020U |
| HWAG6POL\_CNT\_U32 | Bitfield Mask | NA | 0x00000040U |
| HWAG7POL\_CNT\_U32 | Bitfield Mask | NA | 0x00000080U |
| HWTQ0POL\_CNT\_U32 | Bitfield Mask | NA | 0x00000100U |
| HWTQ1POL\_CNT\_U32 | Bitfield Mask | NA | 0x00000200U |
| HWTQ2POL\_CNT\_U32 | Bitfield Mask | NA | 0x00000400U |
| HWTQ3POL\_CNT\_U32 | Bitfield Mask | NA | 0x00000800U |
| HWTQ4POL\_CNT\_U32 | Bitfield Mask | NA | 0x00001000U |
| HWTQ5POL\_CNT\_U32 | Bitfield Mask | NA | 0x00002000U |
| HWTQ6POL\_CNT\_U32 | Bitfield Mask | NA | 0x00004000U |
| HWTQ7POL\_CNT\_U32 | Bitfield Mask | NA | 0x00008000U |
| MOTAGMECL0POL\_CNT\_U32 | Bitfield Mask | NA | 0x00010000U |
| MOTAGMECL1POL\_CNT\_U32 | Bitfield Mask | NA | 0x00020000U |
| MOTAGMECL2POL\_CNT\_U32 | Bitfield Mask | NA | 0x00040000U |
| MOTAGMECL3POL\_CNT\_U32 | Bitfield Mask | NA | 0x00080000U |
| MOTAGMECL4POL\_CNT\_U32 | Bitfield Mask | NA | 0x00100000U |
| MOTAGMECL5POL\_CNT\_U32 | Bitfield Mask | NA | 0x00200000U |
| MOTAGMECL6POL\_CNT\_U32 | Bitfield Mask | NA | 0x00400000U |
| MOTAGMECL7POL\_CNT\_U32 | Bitfield Mask | NA | 0x00800000U |
| MOTELECMECLPOL\_CNT\_U32 | Bitfield Mask | NA | 0x01000000U |
| ASSIMECHPOL\_CNT\_U32 | Bitfield Mask | NA | 0x02000000U |

## Global

|  |
| --- |
| Constant Name |
|  |

## Module specific Lookup Tables Constants

|  |  |  |  |
| --- | --- | --- | --- |
| Constant Name | Resolution | Value | Software Segment |
| None |  |  |  |

# Software Module Implementation

## Sub-Module Functions

## Initialization Functions

*PolarityCfgInit*

## INIT: PolarityCfgInit

## Design Rationale

*Design follows implemenetation in FDD.*

## Module Outputs

*Refer ‘PolarityCfgInit’ block in FDD*

## Module Internal

None

## PERIODIC FUNCTIONS

None

## Interrupt Functions

*None*

## Server runnables

## PolarityCfgRead

## Design Rationale

*None*

## Store Module Inputs to Local copies

*None*

## (Processing of function)………

*Refer ‘PolarityCfgRead’ block in FDD*

## Store Local copy of outputs into Module Outputs

*None*

## PolarityCfgWr

## Design Rationale

*None*

## Store Module Inputs to Local copies

*None*

## (Processing of function)………

*Refer*  ‘*PolarityCfgWr’ block in FDD*

## Store Local copy of outputs into Module Outputs

*None*

## Local Function/Macro Definitions

## Local Function #1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | GetPolarity | Type | Min | Max |
| **Arguments Passed** | Polarity\_Cnt\_T\_u32 | uint32 | 0 | 0xFFFFFFFF |
|  | PolarityMask\_Cnt\_T\_u32 | uint32 | 0x00000001 | 0x02000000 |
| **Return Value** | Polarity\_Cnt\_T\_s08 | sint08 | -1 | 1 |

## Description

* **Design:**

if ( (Polarity\_Cnt\_T\_u32 & PolarityMask\_Cnt\_T\_u32) == PolarityMask\_Cnt\_T\_u32 )

set ‘Polarity\_Cnt\_T\_s08’ to ‘1’

else

set ‘Polarity\_Cnt\_T\_s08’ to ‘-1’

* **Note:**  ‘PolarityMask\_Cnt\_T\_u32’ is a bit field mask and takes values mentioned in table at sec 6.1.1.1

## GLObAL Function/Macro Definitions

None

## Transition FUNCTIONS

None

# Known Limitations With Design

None

# UNIT TEST CONSIDERATION

None

# Appendix A Abbreviations and Acronyms

|  |  |
| --- | --- |
| Abbreviation | Description |
| DFD | Design functional diagram |
| MDD | Module design Document |
| FDD | Functional Design Document |

# Appendix B 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 |  |

# Appendix C References

| **Ref. #** | **Title** | **Version** |
| --- | --- | --- |
| 1 | AUTOSAR Specification of Memory Mapping | v1.3.0 R4.0 Rev 2 |
| 2 | MDD Guideline | EA4 01.00.01 |
| 3 | [Software Naming Conventions.doc](http://misagweb01.nexteer.com/eRoomReq/Files/erooms8/NextGeneration/0_fc55f/Software%20Naming%20Conventions%2003x(In%20Work).doc) | EA4 01.00.00 |
| 4 | Software Design and Coding Standards.doc | 2.1 |
| 5 | FDD : ES102A\_PolarityCfg\_Design | See Synergy sub project version |

# 