**Module Design Document**

**For**

**XCP Interface (XcpIf)**

**VERSION: 4.0**

**DATE: 12-Feb-2018**

**Prepared By:**

**Kevin Smith**

**EPS Software,**

**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** | **Approved By** |
| 1 | Initial Version | K. Smith | 1.0 | 16-Jun-15 |  |
| 2 | Updates for anomaly EA4#6672 | K. Smith | 2.0 | 29-Aug-16 |  |
| 3 | Updates for change request EA4#8215 | K. Smith | 3.0 | 03-Mar-17 |  |
| 4 | Updates for CR EA4#20563 anomaly EA4#20564 | K. Smith | 4.0 | 12-Feb-18 |  |

**Table of Contents**

[1 Abbrevations And Acronyms 5](#_Toc506219806)

[2 References 6](#_Toc506219807)

[3 XCP Interface & High-Level Description 7](#_Toc506219808)

[4 Design details of software module 8](#_Toc506219809)

[4.1 Graphical representation of XCP Interface 8](#_Toc506219810)

[4.2 Data Flow Diagram 8](#_Toc506219811)

[4.2.1 Module level DFD 8](#_Toc506219812)

[4.2.2 Sub-Module level DFD 8](#_Toc506219813)

[4.3 COMPONENT FLOW DIAGRAM 8](#_Toc506219814)

[5 Variable Data Dictionary 9](#_Toc506219815)

[5.1 User defined typedef definition/declaration 9](#_Toc506219816)

[5.2 Variable definition for enumerated types 9](#_Toc506219817)

[6 Constant Data Dictionary 10](#_Toc506219818)

[6.1 Program(fixed) Constants 10](#_Toc506219819)

[6.1.1 Embedded Constants 10](#_Toc506219820)

[6.1.1.1 Local 10](#_Toc506219821)

[6.1.1.2 Global 10](#_Toc506219822)

[6.1.2 Module specific Lookup Tables Constants 10](#_Toc506219823)

[7 Software Module Implementation 11](#_Toc506219824)

[7.1 Sub-Module Functions 11](#_Toc506219825)

[7.2 Initialization Functions 11](#_Toc506219826)

[7.2.1 CDD\_XcpIfInit1 11](#_Toc506219827)

[7.2.1.1 Design Rationale 11](#_Toc506219828)

[7.2.1.2 Store Module Inputs to Local copies 11](#_Toc506219829)

[7.2.1.3 (Processing of function)……… 11](#_Toc506219830)

[7.2.1.4 Store Local copy of outputs into Module Outputs 11](#_Toc506219831)

[7.3 PERIODIC FUNCTIONS 11](#_Toc506219832)

[7.3.1 CDD\_XcpIfPer1 11](#_Toc506219833)

[7.3.1.1 Design Rationale 11](#_Toc506219834)

[7.3.1.2 Store Module Inputs to Local copies 11](#_Toc506219835)

[7.3.1.3 (Processing of function)……… 12](#_Toc506219836)

[7.3.1.4 Store Local copy of outputs into Module Outputs 12](#_Toc506219837)

[7.3.2 Xcp2msDaq 12](#_Toc506219838)

[7.3.2.1 Design Rationale 12](#_Toc506219839)

[7.3.2.2 Store Module Inputs to Local copies 12](#_Toc506219840)

[7.3.2.3 (Processing of function)……… 12](#_Toc506219841)

[7.3.2.4 Store Local copy of outputs into Module Outputs 12](#_Toc506219842)

[7.4 Non PERIODIC FUNCTIONS 12](#_Toc506219843)

[7.5 Interrupt Functions 12](#_Toc506219844)

[7.6 Serial Communication Functions 13](#_Toc506219845)

[7.7 Local Function/Macro Definitions 13](#_Toc506219846)

[7.8 GLObAL Function/Macro Definitions 14](#_Toc506219847)

[7.8.1 ApplXcpGetTimestamp 14](#_Toc506219848)

[7.8.1.1 Description 14](#_Toc506219849)

[7.8.2 ApplXcpGetPointer 14](#_Toc506219850)

[7.8.2.1 Description 14](#_Toc506219851)

[7.8.3 ApplXcpCalibrationWrite 15](#_Toc506219852)

[7.8.3.1 Description 15](#_Toc506219853)

[7.8.4 ApplXcpWrCmn 16](#_Toc506219854)

[7.8.4.1 Description 16](#_Toc506219855)

[7.8.5 ApplXcpCalibrationRead 16](#_Toc506219856)

[7.8.5.1 Description 16](#_Toc506219857)

[7.8.6 ApplXcpCheckWriteAccess 17](#_Toc506219858)

[7.8.6.1 Description 17](#_Toc506219859)

[7.8.7 ApplXcpCheckReadAccess 17](#_Toc506219860)

[7.8.7.1 Description 17](#_Toc506219861)

[7.8.8 ApplXcpCheckDAQAccess 18](#_Toc506219862)

[7.8.8.1 Description 18](#_Toc506219863)

[7.8.9 ApplXcpSetCalPage 19](#_Toc506219864)

[7.8.9.1 Description 19](#_Toc506219865)

[7.8.10 ApplXcpGetCalPage 20](#_Toc506219866)

[7.8.10.1 Description 20](#_Toc506219867)

[7.8.11 ApplXcpCopyCalPage 21](#_Toc506219868)

[7.8.11.1 Description 21](#_Toc506219869)

[7.8.12 ApplXcpUserService 22](#_Toc506219870)

[7.8.12.1 Description 22](#_Toc506219871)

[7.8.13 ApplXcpOpenCmdIf 23](#_Toc506219872)

[7.8.13.1 Description 23](#_Toc506219873)

[7.8.14 NONTRUSTED\_NtWrapS\_Rte\_Call\_CopyCalPageReq\_Oper 24](#_Toc506219874)

[7.8.14.1 Description 24](#_Toc506219875)

[7.8.15 NONTRUSTED\_NtWrapS\_Rte\_Call\_SetCalPageReq\_Oper 24](#_Toc506219876)

[7.8.15.1 Description 24](#_Toc506219877)

[7.9 TRANSIENT FUNCTIONS 25](#_Toc506219878)

[8 Unit Test Considerations 26](#_Toc506219879)

[9 Known Limitations With Design 27](#_Toc506219880)

[10 Appendix 28](#_Toc506219881)

# Abbrevations And Acronyms

|  |  |
| --- | --- |
| Abbreviation | Description |
| DFD | Design functional diagram |
| MDD | Module design Document |
|  | <ADD more to the table if applicable> |
|  |  |

# References

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

|  |  |  |
| --- | --- | --- |
| Sr. No. | Title | Version |
| <1> | <MDD Guidelines> | 4.0.0 |
| <2> | <Software Naming Conventions> | 4.0.0 |
| <3> | <Coding standards> | 4.0.0 |
| <4> | <FDD > | Not available |
|  | <Add if more available> |  |

# XCP Interface & High-Level Description

XCP Interface provides multiple functions that allow XCP end users and tools to interface with software components contained in the application.

# Design details of software module

## Graphical representation of XCP Interface

*None*

## Data Flow Diagram

*None*

## Module level DFD

*None*

## Sub-Module level DFD

*None*

## COMPONENT FLOW DIAGRAM

*None*

# Variable Data Dictionary

## User defined typedef definition/declaration

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Typedef Name | Element Name | User Defined Type | Legal Range  (min) | Legal Range  (max) |
|  |  |  |  |  |
|  |  |  |  |  |

## Variable definition for enumerated types

|  |  |  |
| --- | --- | --- |
| Enum Name | Element Name | Value |
|  |  |  |

# Constant Data Dictionary

## Program(fixed) Constants

## Embedded Constants

## Local

|  |  |  |  |
| --- | --- | --- | --- |
| Constant Name | Resolution | Units | Value |
|  |  |  |  |

## Global

|  |
| --- |
| Constant Name |
| XcpEventChannel\_2ms\_DAQ\_2 |

## Module specific Lookup Tables Constants

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

# Software Module Implementation

## Sub-Module Functions

*None*

## Initialization Functions

## CDD\_XcpIfInit1

## Design Rationale

*This function is called at init to enable or disable XCP access.*

## Store Module Inputs to Local copies

*None*

## (Processing of function)………



## Store Local copy of outputs into Module Outputs

*None*

## PERIODIC FUNCTIONS

## CDD\_XcpIfPer1

## Design Rationale

*This function is called every 100ms to enable or disable XCP access.*

## Store Module Inputs to Local copies

*None*

## (Processing of function)………



## Store Local copy of outputs into Module Outputs

*None*

## Xcp2msDaq

## Design Rationale

*This function is called every 2ms for executing the XcpEvent functions for the 2ms DAQ.*

## Store Module Inputs to Local copies

*None*

## (Processing of function)………



## Store Local copy of outputs into Module Outputs

*None*

## Non PERIODIC FUNCTIONS

*None*

## Interrupt Functions

*None*

## Serial Communication Functions

*None*

## Local Function/Macro Definitions

None

## GLObAL Function/Macro Definitions

## ApplXcpGetTimestamp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpGetTimestamp | Type | Min | Max |
| **Arguments Passed** | None |  |  |  |
|  |  |  |  |  |
| **Return Value** | Timestamp\_Cnt\_T\_u32 | XcpDaqTimestampType | See description | See description |

## Description

This function returns the timestamp that is based on a reference timer. The range of return values vary depending on the configuration of the Xcp Component. The data type can range from a full range of a uint8 to a uint32 value.



## ApplXcpGetPointer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpGetPointer | Type | Min | Max |
| **Arguments Passed** | addr\_ext | vuint8 | 0 | 255 |
|  | addr | vuint32 | 1 | 4294967295 |
| **Return Value** | RtnAddr\_Cnt\_T\_u32 | MTABYTEPTR | 1 | 4294967295 |

## Description

This function takes the extension and address of and returns the physical address of the item.



## ApplXcpCalibrationWrite

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpCalibrationWrite | Type | Min | Max |
| **Arguments Passed** | addr | MTABYTEPTR | 1 | 4294967295 |
|  | size | Vuint8 | 0 | 255 |
|  | data | BYTEPTR | 1 | 4294967295 |
| **Return Value** | XCP\_CMD\_OK | Uint8 | 0 | 0 |

## Description

This function calls the common XCP writing function. For this deisgn, the function call will be translated into a trusted function call.



## ApplXcpWrCmn

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpWrCmn | Type | Min | Max |
| **Arguments Passed** | addr | MTABYTEPTR | 1 | 4294967295 |
|  | size | Vuint8 | 0 | 255 |
|  | data | BYTEPTR | 1 | 4294967295 |
| **Return Value** | XCP\_CMD\_OK | Uint8 | 0 | 0 |

## Description

This function writes the data passed in by the XCP user to the designated address.



## ApplXcpCalibrationRead

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpCalibrationRead | Type | Min | Max |
| **Arguments Passed** | addr | MTABYTEPTR | 1 | 4294967295 |
|  | size | Vuint8 | 0 | 255 |
|  | data | BYTEPTR | 1 | 4294967295 |
| **Return Value** | XCP\_CMD\_OK | Uint8 | 0 | 0 |

## Description

This function reads the data in the designated address and returns it to the XCP user.



## ApplXcpCheckWriteAccess

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpCheckWriteAccess | Type | Min | Max |
| **Arguments Passed** | addr | MTABYTEPTR | 1 | 4294967295 |
|  | Size\_Cnt\_T\_u08 | Vuint8 | 0 | 255 |
| **Return Value** | XCP\_CMD\_OK | Uint8 | 0 | 0 |

## Description

This function checks access for XCP writes. Since the functions in tuning selection management handle the presmissions for writes, this function shall always return a positive response.



## ApplXcpCheckReadAccess

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpCheckReadAccess | Type | Min | Max |
| **Arguments Passed** | addr | MTABYTEPTR | 1 | 4294967295 |
|  | Size\_Cnt\_T\_u08 | Vuint8 | 0 | 255 |
| **Return Value** | XCP\_CMD\_OK | Uint8 | 0 | 0 |

## Description

This function checks access for XCP reads. Since the functions in tuning selection management handle the presmissions for reads, this function shall always return a positive response.



## ApplXcpCheckDAQAccess

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpCheckDAQAccess | Type | Min | Max |
| **Arguments Passed** | addr | MTABYTEPTR | 1 | 4294967295 |
|  | Size\_Cnt\_T\_u08 | Vuint8 | 0 | 255 |
| **Return Value** | XCP\_CMD\_OK | Uint8 | 0 | 0 |

## Description

This function checks access for XCP DAQ access. Since all reads are allowed, this function will also always return a positive response.



## ApplXcpSetCalPage

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpSetCalPage | Type | Min | Max |
| **Arguments Passed** | Seg\_Cnt\_T\_u08 | Vuint8 | 0 | 255 |
|  | Page\_Cnt\_T\_u08 | Vuint8 | 0 | 255 |
|  | Mod\_Cnt\_T\_u08 | Vuint8 | 0 | 255 |
| **Return Value** | Rtn\_Cnt\_T\_u08 | Uint8 | 0 | 0x28 |

## Description

This function sets the calibration page access. It directly calls the functions used in tuning selection management.



## ApplXcpGetCalPage

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpGetCalPage | Type | Min | Max |
| **Arguments Passed** | Seg\_Cnt\_T\_u08 | Vuint8 | 0 | 255 |
|  | Mod\_Cnt\_T\_u08 | Vuint8 | 0 | 255 |
| **Return Value** | Rtn\_Cnt\_T\_u08 | Uint8 | 0 | 0x28 |

## Description

This function sets the calibration page access. It directly calls the functions used in tuning selection management.



## ApplXcpCopyCalPage

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpCopyCalPage | Type | Min | Max |
| **Arguments Passed** | SrcSeg\_Cnt\_T\_u08 | Vuint8 | 0 | 255 |
|  | SrcPage\_Cnt\_T\_u08 | Vuint8 | 0 | 255 |
|  | DestSeg\_Cnt\_T\_u08 | Vuint8 | 0 | 255 |
|  | DestPage\_Cnt\_T\_u08 | Vuint8 | 0 | 255 |
| **Return Value** | XCP\_CMD\_OK | Uint8 | 0 | 0x28 |

## Description

This function sets the calibration page access. It directly calls the functions used in tuning selection management.



## ApplXcpUserService

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpUserService | Type | Min | Max |
| **Arguments Passed** | pCmd | BYTEPTR | 0 | 4294967295 |
| **Return Value** | XCP\_CMD\_OK | Uint8 | 0 | 0x3 |

## Description

This function handles user service requests.



## ApplXcpOpenCmdIf

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | ApplXcpOpenCmdIf | Type | Min | Max |
| **Arguments Passed** | pCmd | BYTEPTR | 0 | 4294967295 |
|  | pRes | BYTEPTR | 0 | 4294967295 |
|  | pLength | BYTEPTR | 0 | 4294967295 |
| **Return Value** | XCP\_CMD\_OK | Uint8 | 0 | 0x3 |

## Description

This function handles XCP service requests that are not supported by the driver but defined by the XCP protocol specification.



## NONTRUSTED\_NtWrapS\_Rte\_Call\_CopyCalPageReq\_Oper

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | NONTRUSTED\_NtWrapS\_Rte\_Call\_CopyCalPageReq\_Oper | Type | Min | Max |
| **Arguments Passed** | FunctionIndex | NonTrustedFunctionIndexType | 0 | 65535 |
|  | FunctionParams | NonTrustedFunctionParameterRefType | N/A | N/A |
| **Return Value** | N/A | N/A | N/A | N/A |

## Description



## NONTRUSTED\_NtWrapS\_Rte\_Call\_SetCalPageReq\_Oper

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function Name** | NONTRUSTED\_NtWrapS\_Rte\_Call\_SetCalPageReq\_Oper | Type | Min | Max |
| **Arguments Passed** | FunctionIndex | NonTrustedFunctionIndexType | 0 | 65535 |
|  | FunctionParams | NonTrustedFunctionParameterRefType | N/A | N/A |
| **Return Value** | N/A | N/A | N/A | N/A |

## Description



## TRANSIENT FUNCTIONS

*None*

# Unit Test Considerations

* The value datatype of XcpDaqTimestampType can be uint8, uint16, or uint32 depending on the configuration of XCP. Unit testing shall test full ranges for all three types to ensure proper functionality.

# Known Limitations With Design

* There are no protections in this design for executing on NULL\_PTRs

# Appendix

*None*