HYUNDAI AUTOEVER

AUTOSAR Crc User Manual

DOC. NO

SCOPE OF APPLICATION All Project/Engineering
Responsibility : Classic AUTOSAR Team

File Name Crc_UM
Creation YJ Yun 2023/04/14
Check HM Kim 2023/04/14
Approval IW Kang 2023/04/14
11th Edition Date: 2023/04/14
Document Management System

HYUNDAI
AutoEver

Document Change Histroy				
Date (YYYY-MM-DD)	Ver.	Editor	Chap	Description (before -> after revision)
2015-03-30	1.0	CY Song		• Initial Creation
2015-11-25	1.1	CY Song	5.1.1	• Modified Crc16Mode, Crc32Mode, Crc8Mode, Crc8H2FMode categories into Changeable
2016-04-05	1.2	CY Song	4.3.1.1	• Change Logs updated
2016-11-23	1.3	CY Song	4.3.1.1	• Change Logs updated
2019-05-07	1.4	YJ Yun	4.3.1.1	• Change Logs updated
2019-10-17	1.5	YJ Yun	4.3.1.1	• Change Logs updated
2020-12-31	1.3.6.0	YJ Yun	4.3.1.1	• Change Logs updated
2021-01-18	1.3.7.0	YJ Yun	4.3.1.1	• Change Logs updated
2021-12-31	1.3.8.0	JH Lim	4.3.1.1	• Change Logs updated
2022-08-19	1.3.9.0	YJ Yun	4.3.1.1	• Change Logs updated
2023-04-14	1.3.9.1	YJ Yun	4.3.1.1	• Change Logs updated

Table of Contents

1. Overview

This document provides references and guidance for users on parameter configuration and system design during using the AUTOSAR platform for CRC. See reference documents for more details.

Each configuration category is defined as follows.

- Changeable (C): Items that can be configured by users
- Fixed (F): Items that cannot be changed by users
- NotSupported (N): Unavailable items

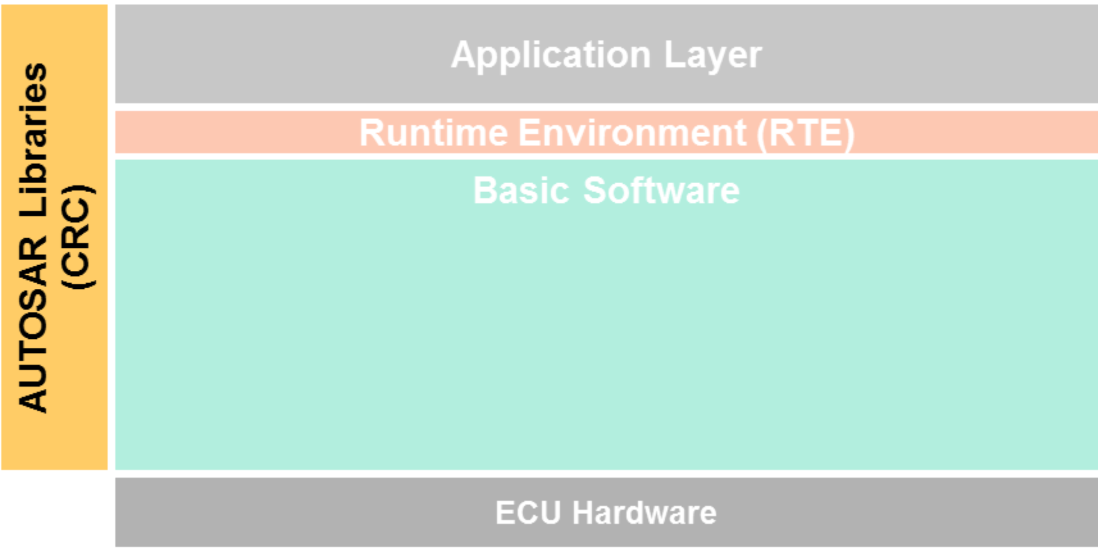
2. Reference

Sl. No.	Title	Version
1.	AUTOSAR_SWS_CRCLibrary.pdf	4.2.0
2.	AUTOSAR BSW Service API Guide.doc	1.0.0 or later

3. AUTOSAR System

3.1 Overview of Software Layers

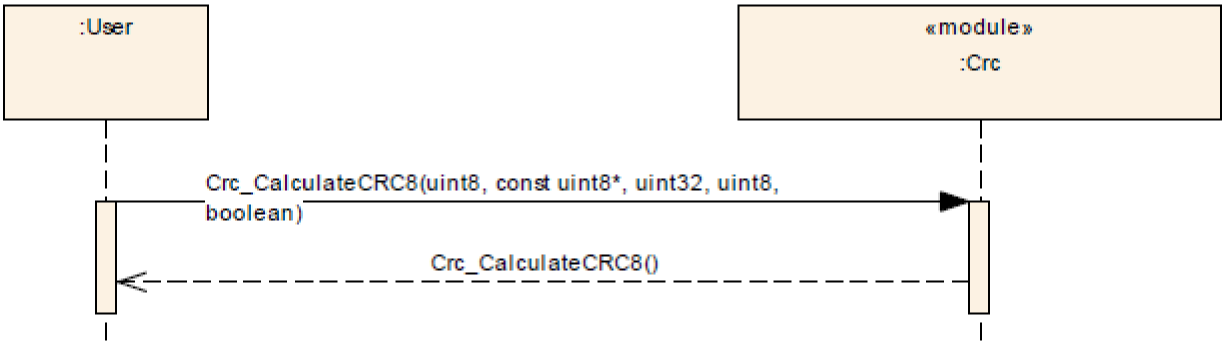
The layered architecture of the AUTOSAR platform related with CRC is illustrated below.



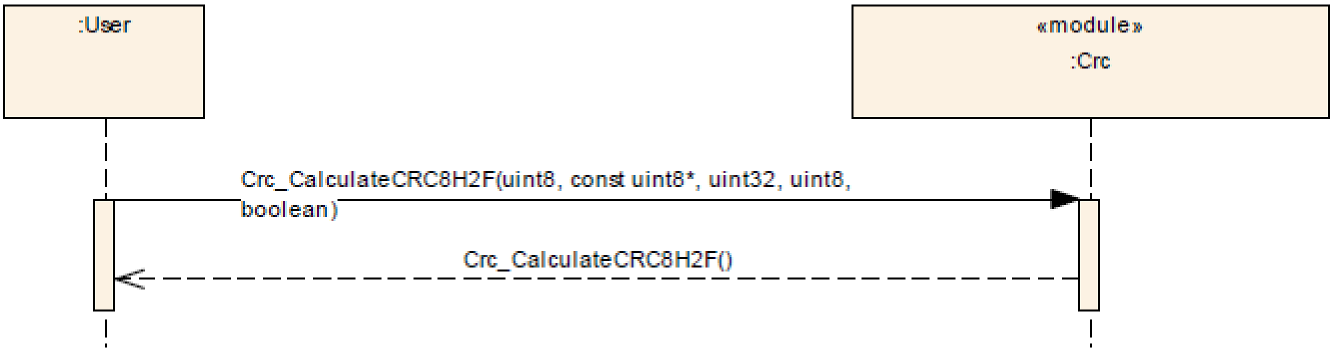
3.2 AUTOSAR CRC Library

3.2.1 Sequence Diagrams

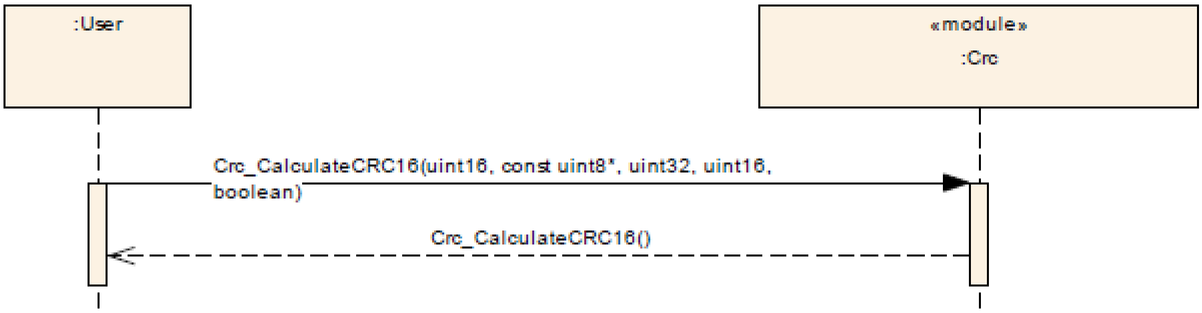
3.2.1.1 CRC_CalculateCRC8()


:User
«module»
:Crc
Crc_CalculateCRC8(uint8, const uint8*, uint32, uint8, boolean)
Crc_CalculateCRC8()

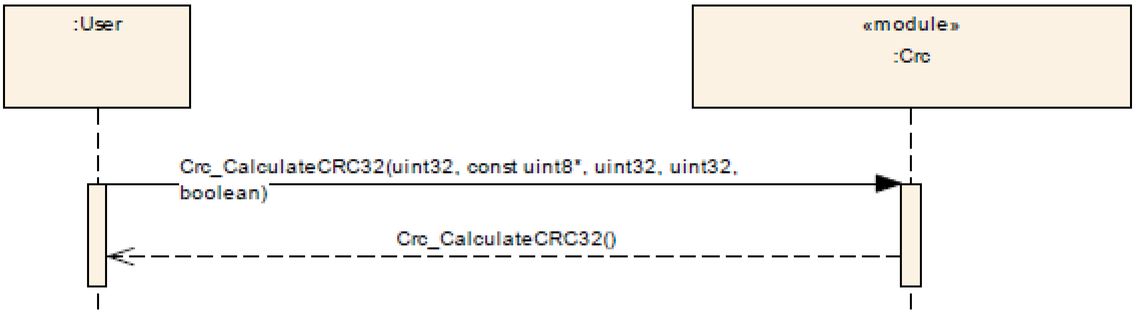
3.2.1.2 Crc_CalculateCRC8H2F ()


:User
«module»
:Crc
Crc_CalculateCRC8H2F(uint8, const uint8*, uint32, uint8, boolean)
Crc_CalculateCRC8H2F()

3.2.1.3 Crc_CalculateCRC16()


:User
«module»
:Crc
Crc_CalculateCRC16(uint16, const uint8*, uint32, uint16, boolean)
Crc_CalculateCRC16()

3.2.1.4 Crc_CalculateCRC32()


:User
«module»
:Crc
Crc_CalculateCRC32(uint32, const uint8*, uint32, uint32, boolean)
Crc_CalculateCRC32()

3.2.2 Architecture (Library)

3.2.2.1 Library can be called by BSW modules(that including the RTE), SW-Cs, libraries or integration code.

3.2.2.2 Library can be re-entrant.

3.2.2.3 Library does not require any initialization.

3.2.2.4 Library are synchronous (they do not have wait points)

4. Product Release Notes

4.1 Overview

The purpose of this chapter is to provide release information about AutoEver's CRC product.
It describes restrictions and noteworthy details of the CRC product's release versions.

4.2 Scope of the release

All content in this document is limited to the following CRC module.

Module	Autosar version	SWS version	Module version
CRC	4.0.3	4.2.0	1.3.9

4.3 Module release notes

4.3.1 CRC

4.3.1.1 Change Log

- **Version 1.3.9.1 (2023-04-14)**
  - Improvements
    - Add the user manual

Cause	Add the user manual English version
Operation effect	None
Setting effect	None
ASW action	None

- **Version 1.3.9.0 (2022-08-20)**
  - Improvements
    - Applied UNECE Rule

Cause	Security coding improved to comply with the UNECE Cyber Security regulations
Operation effect	None
Setting effect	None
ASW action	None

- **Version 1.3.8.0 (2021-12-31)**
  - Improvements
    - Applied MISRA Rule

Cause	Made code improvements to comply with the UNECE Cyber Security regulations
Operation effect	None
Setting effect	None
ASW action	None

➢ Version 1.3.7.0 (2021-01-20)

- Improvements
  - ■ Applied MISRA Rule

Cause	Applied MISRA Rule
Operation effect	None
Setting effect	None
ASW action	None

➢ Version 1.3.6.0 (2020-12-31)

- Improvements
  - ■ Applied MISRA Rule

Cause	Applied MISRA Rule
Operation effect	None
Setting effect	None
ASW action	None

➢ Version 1.3.5.0 (2019-10-17)

- Improvements
  - ■ Modified the directory structure of source files

Cause	Source code made available to partners
Operation effect	None
Setting effect	None
ASW action	None

➢ Version 1.3.5 (2019-05-07)

- Improvements
  - ■ Applied MISRA 2012

Cause	Applied MISRA Rule 2012
Operation effect	None
Setting effect	None
ASW action	None

➢ Version 1.3.4 (2016-06-27)

- Improvements
  - ■ User Manual updated

Cause	User Manual updated

Operation effect	None
Setting effect	None
ASW action	None

- **Version 1.3.3 (2016-04-05)**
  - Added User Manuals to DeliveryBox

4.3.1.2 Limitations

4.3.1.3 Among configuration items, CRC_16_HARDWARE, CRC_32_HARDWARE, CRC_8_HARDWARE, and CRC_8H2F_HARDWARE should be supported by the hardware and not supported by the current CRC module.

4.3.1.4 Deviation

None

5. Configuration Guide

5.1 Crc Module

5.1.1 CrcGeneral Container

See the following configurations.

Parameter Name	Value	Category
Crc16Mode	CRC_16_RUNTIME	C
Crc32Mode	CRC_32_RUNTIME	C
Crc8H2FMode	CRC_8H2F_RUNTIME	C
Crc8Mode	CRC_8_RUNTIME	C

6. Application Programming Interface (API)

6.1 Type Definitions

None

6.2 Macro Constants

None

6.3 Functions

6.3.1 Initialization/Deinitialization Related Api

None

6.3.2 Calcutation of 8bit CRC

Function Name	Crc_CalculateCRC8	
Syntax	FUNC(uint8, CRC_CODE) Crc_CalculateCRC8 (P2CONST(uint8, CRC_CONST, CRC_APPL_CONST)Crc_DataPtr, uint32 Crc_Length, uint8 Crc_StartValue8, boolean Crc_IsFirstCall)	
Service ID	0x01	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	Crc_DataPtr	Pointer to the start address of the data block to be calculated.
	Crc_Length	Length of the data block to calculated in bytes.
	Crc_StartValue8	Initial value when algorithm starts.
	Crc_IsFirstCall	TRUE: First call in a sequence or individual CRC calculation; start from initial value, ignore Crc_StartValue8. FALSE: Subsequent call in a call sequence; Crc_StartValue8 is interpreted to be the return value of the previous function call.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	uint8	8 bit result of CRC calculation.
Description	Calculates CRC (8 bit) using CRC-SAE-J1850 standards by runtime based method. This service makes a CRC8 calculation on Crc_Length data bytes.	
Preconditions	None	
Configuration Dependency	This function shall be configurable with any one of the available CRC8 calculation methods by the configuration parameter : CRC_8_MODE	

6.3.3 Calculation of 8bit CRC with 0X2F polynomial

Function Name	Crc_CalculateCRC8H2F	
Syntax	FUNC(uint8, CRC_CODE) Crc_CalculateCRC8H2F (P2CONST(uint8, CRC_CONST, CRC_APPL_CONST)Crc_DataPtr, uint32 Crc_Length, uint8 Crc_StartValue8H2F, boolean Crc_IsFirstCall)	
Service ID	0x05	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	Crc_DataPtr	Pointer to the start address of the data block to be calculated.

	Crc_Length	Length of the data block to calculated in bytes.
	Crc_StartValue8H2F	Initial value when algorithm starts.
	Crc_IsFirstCall	TRUE: First call in a sequence or individual CRC calculation; start from initial value, ignore Crc_StartValue8H2F. FALSE: Subsequent call in a call sequence; Crc_StartValue8H2F is interpreted to be the return value of the previous function call.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	uint8	8 bit result of CRC calculation.
Description	Calculates CRC (8 bit) using 0x2F polynomial by runtime based method. This service makes a CRC8 calculation with 0x2F polynomial on Crc_Length data bytes.	
Preconditions	None	
Configuration Dependency	This function shall be configurable with any one of the available CRC8H2F calculation methods by the configuration parameter : CRC_8H2F_MODE	

6.3.4 Calculation of 16bit CRC

Function Name	Crc_CalculateCRC16	
Syntax	FUNC(uint16, CRC_CODE) Crc_CalculateCRC16 (P2CONST(uint8, CRC_CONST, CRC_APPL_CONST)Crc_DataPtr, uint32 Crc_Length, uint16 Crc_StartValue16, boolean Crc_IsFirstCall)	
Service ID	0x02	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	Crc_DataPtr	Pointer to the start address of the data block to be calculated.
	Crc_Length	Length of the data block to calculated in bytes.
	Crc_StartValue16	Initial value when algorithm starts.
	Crc_IsFirstCall	TRUE: First call in a sequence or individual CRC calculation; start from initial value, ignore Crc_StartValue16. FALSE: Subsequent call in a call sequence; Crc_StartValue16 is interpreted to be the return value of the previous function call.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	uint16	16 bit result of CRC calculation.
Description	Calculates CRC (16 bit) using CRC-CCITT standards by runtime based method. This service makes a CRC16 calculation on Crc_Length data	

	bytes.
Preconditions	None
Configuration Dependency	This function shall be configurable with any one of the available CRC16 calculation methods by the configuration parameter : CRC_16_MODE

6.3.5 Calculation of 32bit CRC

Function Name	Crc_CalculateCRC32	
Syntax	FUNC(uint32, CRC_CODE) Crc_CalculateCRC32 (P2CONST(uint8, CRC_CONST, CRC_APPL_CONST)Crc_DataPtr, uint32 Crc_Length, uint32 Crc_StartValue32, boolean Crc_IsFirstCall)	
Service ID	0x03	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	Crc_DataPtr	Pointer to the start address of the data block to be calculated
	Crc_Length	Length of the data block to calculated in bytes.
	Crc_StartValue32	Initial value when algorithm starts
	Crc_IsFirstCall	TRUE: First call in a sequence or individual CRC calculation; start from initial value, ignore Crc_StartValue32. FALSE: Subsequent call in a call sequence; Crc_StartValue32 is interpreted to be the return value of the previous function call.
Parameters (Inout)	None	
Parameters (Out)	None	
Return Value	uint32	32 bit result of CRC calculation.
Description	Calculates CRC (32 bit) using CRC-IEEE 802.3 standards by runtime based method. This service makes a CRC32 calculation on Crc_Length data bytes.	
Preconditions	None	
Configuration Dependency	This function shall be configurable with any one of the available CRC32 calculation methods by the configuration parameter : CRC_32_MODE	

6.3.6 GetVersion Information

Function Name	Crc_GetVersionInfo	
Syntax	FUNC(void, CRC_CODE) Crc_GetVersionInfo (P2VAR(Std_VersionInfoType, AUTOMATIC, CRC_APPL_DATA) Versioninfo)	
Service ID	0x04	
Sync/Async	Synchronous	
Reentrancy	Reentrant	
Parameters (In)	None	
Parameters (Inout)	None	
Parameters (Out)	Versioninfo	Pointer represents where the

		version information of this module needs to be stored
Return Value	None	
Description	This function returns the version information of the module, which includes VendorId, ModuleId and Vendor Specific Version numbers. If Versioninfo is a NULL pointer then this service will simply return.	
Preconditions	None	
Configuration Dependency	None	

7. Generator

7.1 Generator Option

None

7.2 Generator Error Message

7.2.1 Crc

7.2.1.1 Error Messages

1) ERR201003: ‘Component Name’ Component is not present in the input file(s).
   - This error occurs, if ‘Crc’ component is not present in any of the input ECU Configuration Description File(s).

2) ERR201005: The parameter ‘Parameter Name’ in the container ‘Container Name’ should be configured.
   - This error occurs, if any of the mandatory configuration parameters mentioned below are not configured in ECU Configuration Description File.

Container Name	Parameter Name
BSW-IMPLEMENTATION	AR-RELEASE-VERSION
	VENDOR-ID
	SW-VERSION
BSW-MODULE-DESCRIPTION	MODULE-ID

3) ERR201006: The value configured for the parameter ‘Parameter Name’ in the container ‘Container Name’ should follow the pattern: <Pattern>.
   - This error occurs, when the parameter ‘Parameter Name’ is not configured as per the pattern.

Parameter Name	Container Name	Pattern	Example
AR-RELEASE-VERSION	BSW-IMPLEMENTATION	<4.[0-9]+.[0-9]+>	4.0.3
SW-VERSION			

7.2.1.2 Warning Messages

None

7.2.1.3 Information Messages

1) INF201015: AUTOSAR Release version <value of the element AR-RELEASE-VERSION> configured for the parameter 'AR-RELEASE-VERSION' in provided MDT file is not correct. AUTOSAR Release version should be one of the following: 4.0.3.

- This information occurs, if the value of the element AR-RELEASE-VERSION present in the BSW Module Description template is configured other than 4.0.3.

2) INF201051: Value of the parameter 'Parameter Name' of the container 'CrcGeneral' is configured as 'Enumeration Literal', hence Generation Tool resets the value of the parameter to 'Reset Value'.

- This information occurs, if the below mentioned parameters are configured as 'Enumeration Literal'.

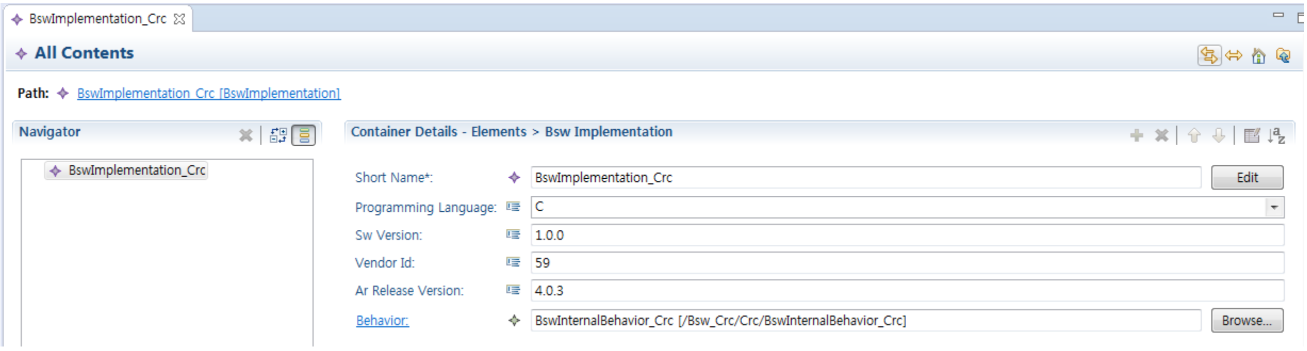
Parameter Name	Enumeration Literal	Reset Value
Crc16Mode	CRC_16_HARDWARE	CRC_16_RUNTIME
Crc32Mode	CRC_32_HARDWARE	CRC_32_RUNTIME
Crc8H2FMode	CRC_8H2F_HARDWARE	CRC_8H2F_RUNTIME
Crc8Mode	CRC_8_HARDWARE	CRC_8_RUNTIME

8. Appendix

8.1 Bswmd (Bsw Module Description)

8.1.1 Bsw Module Version Setting

When compiling each module, incorrect version information triggers a compile error.
This requires version information revision in BswImplementation Container as the following Bswmd.



Path: BswImplementation_Crc [BswImplementation]

Short Name*: BswImplementation_Crc

Programming Language: C

Sw Version: 1.0.0

Vendor Id: 59

Ar Release Version: 4.0.3

Behavior: BswInternalBehavior_Crc [/Bsw_Crc/Crc/BswInternalBehavior_Crc]

8.2 Exclusive Areas

None

8.3 Example

8.3.1 calculation of CRC8

8.3.1.1 Data bytes 01h 02h 03h 04h 05h 06h 07h 08h: start value FFh:

Result = Crc_CalculateCRC8(&Array12345678[0], 8, 0xFF, TRUE);

8.3.2 calculation of CRC8H2F()

8.3.2.1 Data bytes 01h 02h 03h 04h 05h 06h 07h 08h: start value FFh:

Result = Crc_CalculateCRC8H2F(&Array12345678[0], 8, 0xFF, TRUE);

8.3.3 calculation of CRC16()

8.3.3.1 Data bytes 01h 02h 03h 04h 05h 06h 07h 08h: start value FFFFh:

Result = Crc_CalculateCRC16(&Array12345678[0], 8, 0xFFFF, TRUE);

8.3.4 calculation of CRC32()

8.3.4.1 Data bytes 01h 02h 03h 04h 05h 06h 07h 08h: start value FFFFFFFFh:

Result = Crc_CalculateCRC32(&Array12345678[0], 8, 0xFFFFFFFF, TRUE);