


SCOPE OF APPLICATION All Project/Engineering		SHT/SHTS 1 / 19
Responsibility: 클래식오토사1팀	AUTOSAR Data Log User Manual	DOC. NO
<h1>AUTOSAR Data Log User Manual</h1>		

Document Change History				
Date (YYYY-MM-DD)	Ver.	Editor	Chap	내용(개정 전 -> 개정 후)
2020-11-25	1.0.0.0	MJ.Woo	All	• Initial Creation
2020-12-18	1.0.1.0	MJ.Woo	4	• Update release note
2021-06-09	1.1.0.0	MJ.Woo	4 5.3	• Update release note • Add “HwResource 설정”
2021-06-29	1.1.0.1	MJ.Woo	5.2.1	• Fixed DemUdsDTC value
2021-12-27	1.1.1.0	MJ.Woo	All 4.3.1	• Change company name • Update release note
2022-08-09	1.1.2.0	YH.Han	4.3.1	• Update release note
2023-04-11	2.0.0.0	MJ.Woo	All	• Changed the log reading method from Read DTC to ReadMemoryByAddress and reflected the resulting design changes

Edition Date: 2023-04-11	File Name DataLog_UM.pdf	Creation MJ Woo 2023-04-11	Check JH Cho 2023-04-11	Approval JH Jung 2023-04-11
Document Management System				

## Table of Contents

<b>1. OVERVIEW</b>	<b>4</b>
<b>2. REFERENCE</b>	<b>4</b>
<b>3. AUTOSAR SYSTEM</b>	<b>5</b>
<b>3.1 Overview of Software Layers</b>	<b>5</b>
<b>3.2 DataLog</b>	<b>6</b>
3.2.1 수집 정보	6
3.2.2 Custom Data 저장	6
<b>4. PRODUCT RELEASE NOTES</b>	<b>7</b>
<b>4.1 Overview</b>	<b>7</b>
<b>4.2 Scope of the release</b>	<b>7</b>
<b>4.3 Module release notes</b>	<b>7</b>
4.3.1 Change Log	7
4.3.1.1 Version 2.0.0.0	7
4.3.1.2 Version 1.1.2.0	7
4.3.1.3 Version 1.1.0.1	7
4.3.1.4 Version 1.1.0.0	8
4.3.1.5 Version 1.0.1.0	8
4.3.1.6 Version 1.0.0.0	8
4.3.2 Limitation	8
4.3.2.1 Log 저장 제약 사항	8
4.3.2.2 수집 정보 제약	8
4.3.2.3 타 모듈 의존성	9
4.3.2.4 SWP SRS 설정 제약	9
4.3.3 Deviation	9
<b>5. CONFIGURATION GUIDE</b>	<b>10</b>
<b>5.1 NvM 설정</b>	<b>10</b>
5.1.1 NvMBlockDescriptor	10
<b>5.2 HwResource 설정</b>	<b>11</b>
5.2.1 Feature Definition	11
<b>5.3 Dcm 설정</b>	<b>11</b>
<b>6. APPLICATION PROGRAMMING INTERFACE (API)</b>	<b>12</b>
<b>6.1 Type Definitions</b>	<b>12</b>
6.1.1 DataLog_DataType	12
6.1.2 DataLog_EcuMModeType	12

6.1.3	DataLog_SleepSeqType.....	13
<b>6.2</b>	<b>Macro Constants .....</b>	<b>13</b>
<b>6.3</b>	<b>Functions .....</b>	<b>13</b>
6.3.1	DataLog_MainFunction .....	13
6.3.2	DataLog_GetCustomData .....	13
6.3.3	DataLog_SetCustomData.....	14
6.3.4	DataLog_GetNvLog.....	14
6.3.5	DataLog_SetNvLog .....	15
<b>6.4</b>	<b>Global Variables.....</b>	<b>15</b>
<b>7.</b>	<b>GENERATOR.....</b>	<b>16</b>
<b>8.</b>	<b>SWP ERROR CODE .....</b>	<b>16</b>
<b>8.1</b>	<b>Dem Error.....</b>	<b>16</b>
<b>8.2</b>	<b>Det Error.....</b>	<b>16</b>
<b>9.</b>	<b>APPENDIX .....</b>	<b>17</b>
<b>9.1</b>	<b>설계 시 유의사항 .....</b>	<b>17</b>
9.1.1	DataLog 관련 NvM Block 추가 .....	17
9.1.2	ReadMemoryByAddress 주소값 설정 .....	17
<b>9.2</b>	<b>Log 분석.....</b>	<b>17</b>
9.2.1	정상 Log .....	18
9.2.2	에러 Log1 .....	18
9.2.3	에러 Log2 .....	18
9.2.4	에러 Log3 .....	18
9.2.5	에러 Log4 .....	18

## 1. Overview

DataLog 는 SWP 에서 동작하는 Sleep, Wakeup 과정에서의 주요 상태 정보를 수집한다. 이를 기반으로 사용자는 Sleep 실패나 의도치 않은 Wakeup 의 원인을 파악할 수 있다. 또한, Reset Reason 등 추가적인 동작 정보를 수집하여 문제가 발생할 경우 그 이유를 추정할 수 있다.

설정관련 Category 의 해석은 다음과 같다.

- Changeable (C): User 에 의해서 설정 가능한 항목
- Fixed (F): User 에 의한 변경이 불가능한 항목
- NotSupported (N): 사용되지 않는 항목

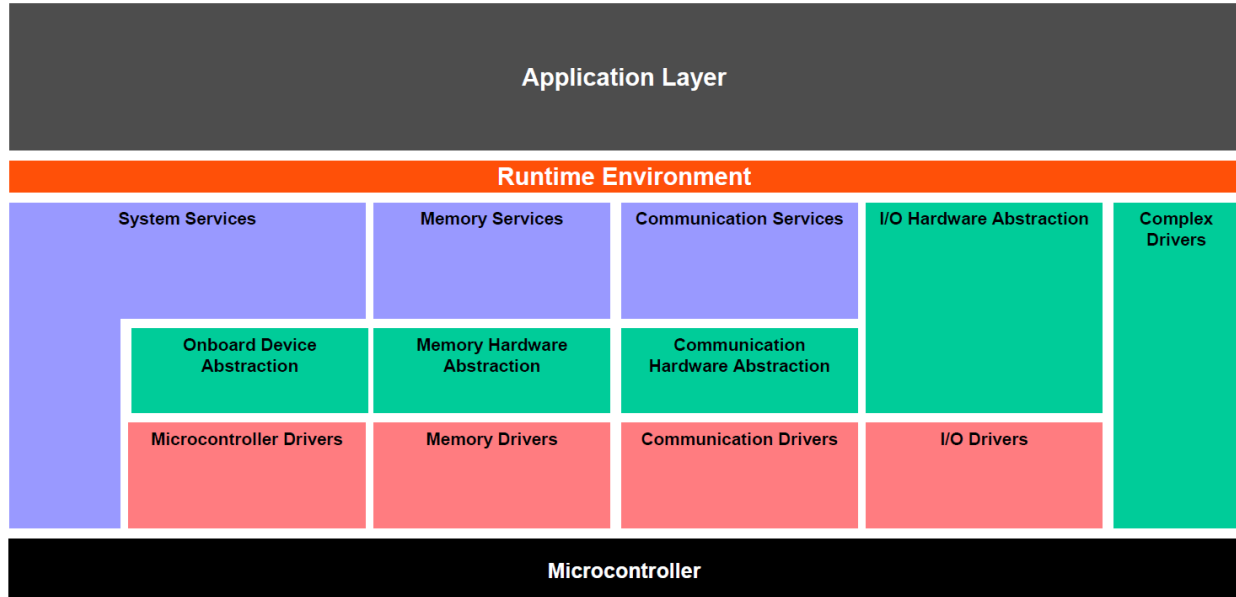
## 2. Reference

Sl. No.	Title	Version
1.	-	
2.	-	

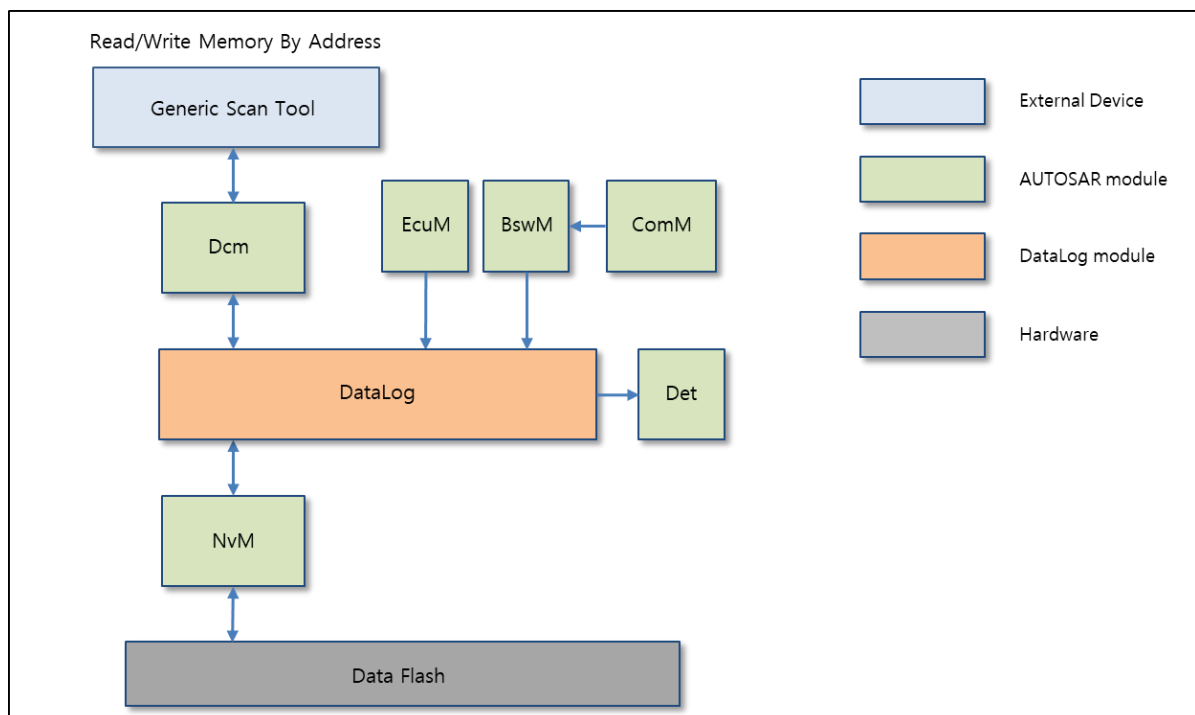
### 3. AUTOSAR System

#### 3.1 Overview of Software Layers

AUTOSAR 플랫폼의 Layered Architecture 는 아래와 같다. AUTOSAR 플랫폼은, Service Layer, ECU Abstraction Layer, Complex Device Drivers 및 Microcontroller Abstraction Layer 로 구분될 수 있다.



DataLog 는 Complex Drivers 영역에 속하며, 하기 그림과 같은 인터페이스를 가진다.



## 3.2 DataLog

### 3.2.1 수집 정보

DataLog 는 SWP 에서 동작하는 Sleep, Wakeup 과정에서의 주요 상태 정보를 수집한다. 여기에서는 DataLog 가 언제 어떤 정보를 수집하는가에 대해 설명한다.

- MainFunction 최초 실행  
Reset Reason 수집
- SLEEP 요청(Rte\_Call\_clientPort\_StateRequest\_RequestSleep() 호출)  
NvM 의 Low Power Allow 상태  
통신 채널의 상태(COMM\_NO\_COMMUNICATION 인지 여부)
- SLEEP 모드 전환 완료(BswM Rule 통과 후 MCU Low Power Mode 전환 전)  
SLEEP 조건을 만족하여 BswM Rule 을 통과하였는지 여부
- Wakeup 후  
Wakeup Event 정보
- Wakeup 후 MainFunction 실행  
Wakeup 후 RUN 상태로 정상 전환되었는지 여부

위 정보들은 Sleep, Wakeup 과정에서 오류가 있다고 판단될 때만 NvM 에 저장된다.  
저장된 Log 에서 문제를 파악하는 방법은 Appendix 의 “9.2 Log 분석”을 참조.

### 3.2.2 Custom Data 저장

DataLog 는 자체적으로 수집하는 Sleep/Wakeup Log 외에 사용자의 Custom Data 를 저장하는 기능을 제공한다.

저장은 DataLog\_SetCustomData() API 를 이용한다. (6.3.2 참조)

Custom Data 의 사이즈는 2byte 이다.

## 4. Product Release Notes

### 4.1 Overview

이 Chapter 에서는, DataLog 에 대한 release 관련 내용을 제공하는데 목적이 있으며, DataLog Software product release version 에 대한, 제한사항 및 특이사항을 기술하고 있다.

### 4.2 Scope of the release

이 문서에 대한 모든 내용은, 다음의 DataLog 버전에 한정한다.

Module	Module version
DataLog	2.0.0

※ Module version 은 각 모듈의 BswModule Description(Bswmd)파일의 Sw version 을 의미한다.

### 4.3 Module release notes

#### 4.3.1 Change Log

##### 4.3.1.1 Version 2.0.0.0

##### ➤ 개선 사항

##### ■ Read DTC 에서 Read Memory By Address 로 저장 정보 확인방법 변경

• 원인	• 디버깅 정보가 DTC 오류로 오인될 수 있음
• 동작 영향	• 없음
• 설정 영향	• NvM Block 추가. Dem 내 DataLog 설정 제거. • Dcm Read/Write Memory By Address 설정
• ASW 조치 필요 사항	• 없음

##### 4.3.1.2 Version 1.1.2.0

##### ➤ 개선 사항

##### ■ UNECE Cyber Security 법규 대응을 위한 보안 코딩 개선

• 원인	• UNECE Cyber Security 대응 필요
• 동작 영향	• 없음
• 설정 영향	• 없음
• ASW 조치 필요 사항	• 없음

##### 4.3.1.3 Version 1.1.0.1

##### ➤ 개선 사항

##### ■ DataLog 용 UDS DTC 값 고정

• 원인	• DataLog 용으로 할당된 DTC 값 반영 필요
• 동작 영향	• 없음
• 설정 영향	• 없음
• ASW 조치 필요 사항	• DataLog DemUdsDTC 설정값 0xDD0000 으로 변경

**4.3.1.4 Version 1.1.0.0**

## ➤ 신규 기능

## ■ ON/OFF 기능 제공

• 원인	• DataLog 를 포함한 SWP 배포 후 기능 비활성화 필요
• 동작 영향	• 없음
• 설정 영향	• 없음
• ASW 조치 필요 사항	• 없음

## ➤ 개선 사항

## ■ Rte R44 와의 호환 지원

• 원인	• Rte R40 과 R44 의 DataLog 타입 생성 위치 불일치
• 동작 영향	• 없음
• 설정 영향	• 없음
• ASW 조치 필요 사항	• 없음

**4.3.1.5 Version 1.0.1.0**

## ➤ 개선 사항

## ■ RamTst 의 Low Power Allow 상태 오류를 탐지 못하는 문제 수정

• 원인	• Preprocessor 체크 오류로 RamTst 의 Low Power Allow 상태 오류를 탐지 못함
• 동작 영향	• 없음
• 설정 영향	• 없음
• ASW 조치 필요 사항	• 없음

**4.3.1.6 Version 1.0.0.0**

## ➤ 신규 기능

## ■ 최초 버전

• 원인	• DataLog 기능 요구
• 동작 영향	• 없음
• 설정 영향	• 없음
• ASW 조치 필요 사항	• 없음

**4.3.2 Limitation****4.3.2.1 Log 저장 제약 사항**

- Data Flash 저장 시점: DataLog는 NvM을 이용해 Data Flash 에 수집한 Log를 저장한다. 따라서 NvM이 동작하지 않을 때에는 Data Flash에 Log를 기록할 수 없다.

**4.3.2.2 수집 정보 제약**

- 수집 가능한 통신 채널(ComM Channel)은 0 ~ 31의 32개다. 그 이상의 통신 채널은 수집 대상에서 제외되고 SLEEP 요청 시 오류 파악도 불가하다.



#### 4.3.2.3 타 모듈 의존성

- DataLog에서는 AUTOSAR specification에 정의된 API 외에 AUTOEVER 구현 모듈의 내부 데이터를 사용하므로 다음 모듈들은 AUTOEVER 외의 타사 모듈을 사용할 수 없다.

- NvM

#### 4.3.2.4 SWP SRS 설정 제약

- Dcm 항목에서 “ReadMemoryByAddress” 기능을 사용해야 저장된 로그를 진단기로 확인할 수 있다.
- Dcm 항목에서 “WriteMemoryByAddress” 기능을 사용해야 저장된 로그를 진단기로 초기화 할 수 있다.

#### 4.3.3 Deviation

None

## 5. Configuration Guide

### 5.1 NvM 설정

DataLog 를 사용하기 위해 필요한 NvM 필수 설정을 설명한다. 자세한 설명은 NvM User Manual 참고.

#### 5.1.1 NvMBlockDescriptor

Parameter Name	Value	Category
Short Name	NvMBlock_DataLog	F
NvMNvBlockLength	4	F
NvMRamBlockDataAddress	NvM_R40: DataLog_NvLog NvM_R44: &DataLog_NvLog	F
NvMSelectBlockForReadAll	True	F
NvMSelectBlockForWriteAll	False	C

- 1) Short Name: DataLog 에서 사용하는 NvM Block 은 반드시 NvMBlock\_DataLog 를 이름으로 지정해야 한다.
- 2) NvMNvBlockLength: DataLog 에서 사용하는 NvM Block 의 크기는 4 byte 로 고정이다.
- 3) NvMRamBlockDataAddress: DataLog 내부 변수인 DataLog\_NvLog 를 지정한다.
- 4) NvMSelectBlockForReadAll: DataLog 는 Startup 시 NvM\_ReadAll()에서 저장된 정보를 읽는다고 가정한다.
- 5) NvMSelectBlockForWriteAll: DataLog 는 MainFunction 에서 저장될 정보의 변경이 있을 때마다 NvM 에 저장을 요청하므로 기본값은 'False'이다.

\* 그 외 설정은 mobilgene C Studio 'Memory Block Editor'의 기본 설정을 사용한다.

## 5.2 HwResource 설정

### 5.2.1 Feature Definition

Parameter Name	Value	Category
SWP_USE_DATALOG	STD_ON	C

- 1) SWP\_USE\_DATALOG
  - DataLog 사용: STD\_ON
  - DataLog 미사용: STD\_OFF

## 5.3 Dcm 설정

기본 설정은 아래 링크를 참조한다:

[\[Dcm\]\[R40\]\[SAG\] WriteMemoryByAddress / ReadMemoryByAddress 사용 방법](#)

ReadMemoryByAddress 와 WriteMemoryByAddress 서비스로 사용할 DataLog 의 메모리 공간은 Linker Directive File 에 “DATALOG\_RMBA\_AREA” 라는 section 이름으로 배치되어 있다.

이 위치를 변경할 경우 LD 파일과 함께 아래 경로에서 메모리 주소를 같이 변경해 주어야 한다.

- Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/Id Info/MemoryIdInfo/Read Memory Range Info
- Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/Id Info/MemoryIdInfo/Write Memory Range Info

## 6. Application Programming Interface (API)

### 6.1 Type Definitions

#### 6.1.1 DataLog\_DataType

<b>Name:</b>	DataLog_DataType		
<b>Type:</b>	struct		
<b>Range:</b>	uint32	ComMMode	ComM Mode status - Bit position: Channel ID, - Value: 1 - FULL_COM or SILENT_COM 0 - NO_COM
	boolean	LpAllowNvM	LowPowerAllow status of NvM - Value: TRUE - ALLOW, FALSE - PREVENT
	DataLog_EcuMModeType	EcuMMode	ECU state
	EcuM_WakeupSourceType	WakeupEvent	Source of Wakeup Event
	boolean	SleepOK	BswM SLEEP condition passed
	boolean	RunOK	ECU RUN status
	boolean	ViolationComMMode	ComM Mode condition violation
	boolean	ViolationNvMMode	NvM condition violation
	DataLog_SleepSeqType	SleepSequence	Current Sleep and Wakeup sequence
	Mcu_ResetType	ResetReason	Reset Reason
<b>Description:</b>	Types for Log Data		

#### 6.1.2 DataLog\_EcuMModeType

<b>Name:</b>	DataLog_EcuMModeType		
<b>Type:</b>	enum		
<b>Range:</b>	DATALOG_ECUM_STARTUP	0	
	DATALOG_ECUM_STARTUP_ONE	1	
	DATALOG_ECUM_STARTUP_TWO	2	
	DATALOG_ECUM_STARTUP_THREE	3	
	DATALOG_ECUM_RUN	4	
	DATALOG_ECUM_POST_RUN	5	
	DATALOG_ECUM_SHUTDOWN	6	
	DATALOG_ECUM_PREP_SHUTDOWN	7	
	DATALOG_ECUM_SLEEP	8	
	DATALOG_ECUM_OFF	9	
	DATALOG_ECUM_RESET	10	
<b>Description:</b>	This type indicates EcuM Mode		

## 6.1.3 DataLog\_SleepSeqType

<b>Name:</b>	DataLog_SleepSeqType		
<b>Type:</b>	enum		
<b>Range:</b>	DATALOG_SEQ_INIT	0x00	From Power On Reset to before ECU RUN
	DATALOG_SEQ_SLEEP_REQ	0x01	SLEEP was requested
	DATALOG_SEQ_SLEEP_DELAYED	0x02	SLEEP was requested but not performed
	DATALOG_SEQ_H2L	0x03	SLEEP condition of BswM has been passed
	DATALOG_SEQ_L2H	0x04	Wakeup and getting out of SLEEP
	DATALOG_SEQ_RUN_DELAYED	0x05	Wakeup but state transition to RUN fail
	DATALOG_SEQ_RUN	0x06	Normal ECU RUN state
<b>Description:</b>	This type indicates Sleep and Wakeup sequence		

## 6.2 Macro Constants

None

## 6.3 Functions

### 6.3.1 DataLog\_MainFunction

<b>Function Name</b>	DataLog_MainFunction
<b>Syntax</b>	FUNC(void, DATALOG_CODE) DataLog_MainFunction(void)
<b>Service ID</b>	0x00
<b>Sync/Async</b>	Synchronous
<b>Reentrancy</b>	Non Reentrant
<b>Parameters (In)</b>	None
<b>Parameters (Inout)</b>	None
<b>Parameters (Out)</b>	None
<b>Return Value</b>	None
<b>Description</b>	It works periodically to check for errors in Sleep/Wakeup related logs and changes in CustomData and save them to NvM. The period of this function is 100ms.
<b>Preconditions</b>	NvM_ReadAll() should be done
<b>Configuration Dependency</b>	None

### 6.3.2 DataLog\_GetCustomData

<b>Function Name</b>	DataLog_GetCustomData
<b>Syntax</b>	FUNC(uint16, DATALOG_CODE) DataLog_GetCustomData(void)
<b>Service ID</b>	0x09
<b>Sync/Async</b>	Synchronous
<b>Reentrancy</b>	Reentrant
<b>Parameters (In)</b>	None
<b>Parameters (Inout)</b>	None
<b>Parameters (Out)</b>	None
<b>Return Value</b>	uint16 CustomData
<b>Description</b>	Get CustomData from NvLog
<b>Preconditions</b>	None
<b>Configuration Dependency</b>	None

### 6.3.3 DataLog\_SetCustomData

<b>Function Name</b>	DataLog_SetCustomData
<b>Syntax</b>	FUNC(void, DATALOG_CODE) DataLog_SetCustomData(uint16 customData)
<b>Service ID</b>	0x0A
<b>Sync/Async</b>	Synchronous
<b>Reentrancy</b>	Reentrant
<b>Parameters (In)</b>	customData - Data to be saved
<b>Parameters (Inout)</b>	None
<b>Parameters (Out)</b>	None
<b>Return Value</b>	void
<b>Description</b>	This function sets CustomData to NvLog
<b>Preconditions</b>	None
<b>Configuration Dependency</b>	None

### 6.3.4 DataLog\_GetNvLog

<b>Function Name</b>	DataLog_GetNvLog
<b>Syntax</b>	FUNC(uint32, DATALOG_CODE) DataLog_GetNvLog(void)
<b>Service ID</b>	0x0B
<b>Sync/Async</b>	Synchronous
<b>Reentrancy</b>	Reentrant
<b>Parameters (In)</b>	None
<b>Parameters (Inout)</b>	None
<b>Parameters (Out)</b>	None
<b>Return Value</b>	uint32 NvLog
<b>Description</b>	Get NvLog(with custom data). If the return value is 0, it means that there is no error to read from the DataLog.
<b>Preconditions</b>	None

<b>Configuration Dependency</b>	None
---------------------------------	------

### 6.3.5 DataLog\_SetNvLog

<b>Function Name</b>	DataLog_SetNvLog
<b>Syntax</b>	FUNC(void, DATALOG_CODE) DataLog_SetNvLog(uint32 nvlog)
<b>Service ID</b>	0x0C
<b>Sync/Async</b>	Synchronous
<b>Reentrancy</b>	Reentrant
<b>Parameters (In)</b>	nvlog : Data to be saved
<b>Parameters (Inout)</b>	None
<b>Parameters (Out)</b>	None
<b>Return Value</b>	void
<b>Description</b>	This function sets NvLog(with custom data)
<b>Preconditions</b>	None
<b>Configuration Dependency</b>	None

## 6.4 Global Variables

Name:	Type:	Description:
DataLog_Data	DataLog_DataType	Data currently collected
DataLog_NvLog	uint32	Saved Log Data in NvM

## 7. Generator

None

## 8. SWP Error Code

### 8.1 Dem Error

None

### 8.2 Det Error

Type or error	Relevance	Related error code	Value
SLEEP state transition is blocked	Development	DATALOG_E_SLEEP_DELAY	0x01
RUN state transition after wakeup is blocked	Development	DATALOG_E_WAKEUP_DELAY	0x02
Requesting SLEEP while not in COMM_NO_COMMUNICATION	Development	DATALOG_E_COMM_MODE	0x03
Requesting SLEEP in the NvM Low Power Prevent state	Development	DATALOG_E_NVM_MODE	0x04
ComM channel id is greater than 31	Development	DATALOG_E_COMM_CHANNEL_OUT_OF_RANGE	0x05



## 9. Appendix

### 9.1 설계 시 유의사항

#### 9.1.1 DataLog 관련 NvM Block 추가

DataLog 모듈의 추가, 삭제, 업데이트에서 NvM block 이 변경될 수 있다. 이 경우 진단기를 통한 리프로그래밍 /OTA 등으로 업데이트 시 DataFlash 레이아웃이 안 맞아 Fail 이 발생한다.

따라서 DataFlash Erase 후 Flash 1 회 수행이 필요하다. (특히 개발 중인 차량에 업데이트 시 주의 필요)

- 참고: NvM User Manual 의 “9.2.9 Memory Layout 변경시 Erase All”

#### 9.1.2 ReadMemoryByAddress 주소값 설정

ReadMemoryByAddress 를 위한 주소값은 LD 파일을 통해 확인이 가능하며, 해당 주소값이 Ecud\_Dcm 설정값으로 입력된다.

주소값을 변경하고자 하는 경우 기존 ASW 에서 할당하여 사용하는 Ram 영역이나, RamTst (Ecud\_RamTst)의 영역과 겹치지 않도록 주의해야 한다.

### 9.2 Log 분석

아래 표는 DataLog 가 저장하는 Log(DataLog\_NvLog)의 bit 별 의미를 나타낸다.

Bits	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Custom Data															
Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Reserved												NvM Mode Error	ComM Mode Error	Wakeup Delay	Sleep Delay

Field	Description
31-16	Custom Data Data stored separately by the user
15-4	Reserved
3	NvM Mode Error 1b – Requesting SLEEP in the NvM Low Power Prevent state 0b – No error
2	ComM Mode Error 1b – Requesting SLEEP while not in COMM_NO_COMMUNICATION 0b – No error
1	Wakeup Delay 1b – RUN state transition after wakeup is blocked 0b – No error
0	Sleep Delay 1b – SLEEP state transition is blocked 0b – No error

다음은 시나리오별로 나올 수 있는 Log 의 예 이다.

## 9.2.1 정상 Log

Sleep, Wakeup 이 정상적으로 이루어진 경우

Field	Value
NvM Mode Error	0
ComM Mode Error	0
Wakeup Delay	0
Sleep Delay	0

## 9.2.2 예러 Log1

모든 조건 만족이 아닌 상태에서 Sleep 요청. 5 초 이내에 조건 만족하여 Sleep 진입.

항목	값
NvM Mode Error	1 or 0
ComM Mode Error	1 or 0
Wakeup Delay	0
Sleep Delay	0

NvM Mode Error 또는 ComM Mode Error 가 0 이 아니다.

## 9.2.3 예러 Log2

모든 조건 만족이 아닌 상태에서 Sleep 요청. 일정 시간(5s)이 지나도 Sleep 진입 불가.

항목	값
NvM Mode Error	1 or 0
ComM Mode Error	1 or 0
Wakeup Delay	0
Sleep Delay	1

NvM Mode Error 또는 ComM Mode Error 가 0 이 아니다. SLEEP 진입을 못하므로 Sleep Delay 가 1 이다.

## 9.2.4 예러 Log3

Wakeup 했으나 500ms 가 지나도 RUN 전환 불가

항목	값
NvM Mode Error	0
ComM Mode Error	0
Wakeup Delay	1
Sleep Delay	0

SLEEP 진입은 정상적으로 했으나 RUN 전환이 불가하므로 Sleep Delay 는 0, Wakeup Delay 는 1 이다.

DataLog\_Data.WakeupEvent 가 제대로 기록되었다면 AppMode\_WakeupEventValidated() 에서의 처리가 누락되었을 가능성이 크다. WakeupEvent 가 기록되지 않았다면 등록되지 않은 Wakeup Source 나 예외상황으로 인해 MCU Low Power Mode 로의 전환이 실패가 원인일 수 있다.

## 9.2.5 예러 Log4

RUN 상태가 아닐 때 SLEEP 요청

항목	값
NvM Mode Error	0
ComM Mode Error	0
Wakeup Delay	0
Sleep Delay	1

RUN 상태가 아니면 SLEEP 요청을 해도 진입을 못한다. NvM, ComM 같은 다른 조건에서 에러가 없지만 Sleep Delay 가 0 이 된다.