User's Manual

RL78 Family

DALI-2 Control Gear Library
User's Manual: LED (207)

16-bit single chip microprocessor

Renesas Electronics
www.renesas.com

Rev.1.01 Nov. 2022

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.
(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin
   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

How to Use This Manual

1. Purpose and Target Readers

This manual is intended for users who want to develop Control Gear for DALI systems with RL78 microcontrollers.

Basic knowledge of electrical circuits, logic circuits, and microcomputers is required to use this manual.

This manual is broadly categorized and consists of product overview, specifications, and usage instructions.

> Particular attention should be paid to the precautionary notes when using the manual. These notes occur within the body of the text, at the end of each section, and in the Usage Notes section.

> The revision history summarizes the locations of revisions and additions. It does not list all revisions. Refer to the text of the manual for details.

The following documents apply to the DALI Library. Make sure to refer to the latest versions of these documents. The newest versions of the documents listed may be obtained from the Renesas Electronics Web site.

Document Type	Description	Document Title	Document No.
User's Manual Hardware	Hardware specifications (pin layout, memory map, peripheral function specifications, electrical characteristics, timing) and operation description	RL78/I1A User's Manual Hardware	R01UH0169EJ0320
User's Manual Software	Description of CPU instruction set	RL78/I1A User's Manual Software	R01US0015EJ0220
Application note	How to use peripheral functions, application examples Reference programs How to create programs in C language	The information is available on the Renesas Electronics website.	
Renesas Technical Update	Breaking news on product specifications, documents, etc.		

2. Explanation of abbreviations

Abbreviation	English name	Remarks
DALI	Digital Addressable Lighing Interface	International Standard for Lighting Control
NVM	Non-Volatile Memory	Non-volatile memory

Table of Contents

1. DALI207 Library Overview

1.1 Overview of library features

This library is an extension library dedicated to the DALI102 library, which realizes processing of the hardware-independent part of the DALI102 standard as a library for the slave (Control Gear) in DALI communication.
For the specifications of the DALI102 library, refer to the DALI102 Library User's Manual.
This library provides the hardware-independent part of the specifications specified in IEC62386-207ed1.0 (hereinafter referred to as DALI207) and should be used when using LEDs as Control Gear lamps and when extending the functionality of LED lamps.

Table 1.1 Processing range

User creation processing	Library processing
・Dimming control (for high-speed fade) ・LED lamp failure detection	・Received 16-bit forward frame processing (partial) ・Transmitted backward frame issuance (partial) ・Timing control ・DALI variable manipulation

This library provides extensions to each logical unit defined using the DALI102 library. The extended logical units will automatically process DALI207 commands with the receive command processing functions provided by the DALI102 library.
There are various commands that can be specified in the 16-bit forward frame, such as DALI variable setting commands and DALI variable setting value acquisition commands. If it is necessary to change the settings on the application side, the application will be notified as necessary.

1.2 Software configuration

The Control Gear software configuration when using this library is shown below.
The part surrounded by the red line is this library. This library is assumed to be extended to the DALI102 library.
In addition, this library can be extended to the DALI209 library, which is released separately for the DALI102 library.

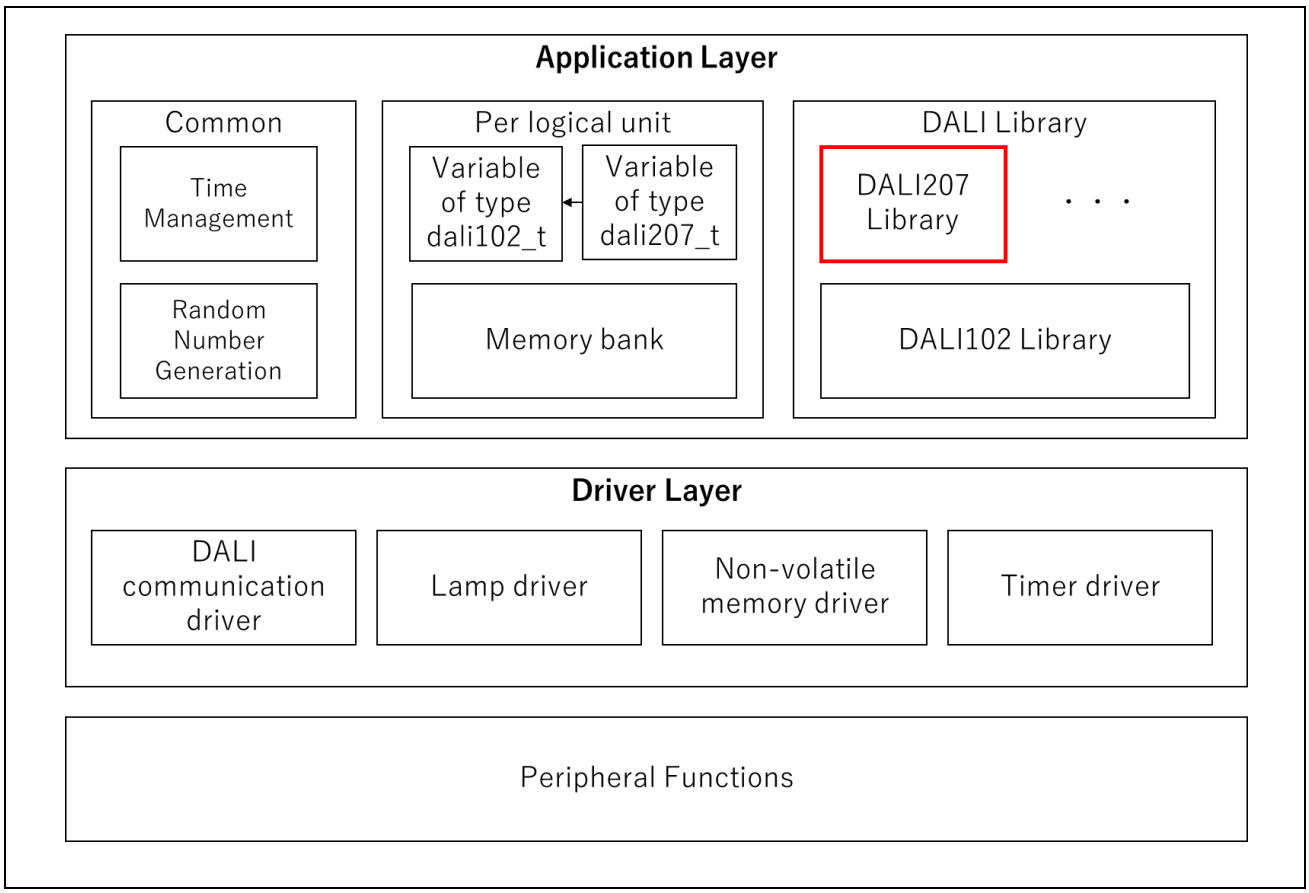


Figure 1.1 Control Gear software configuration diagram

1.3 Supported standard

The standards supported by this library are as follows.

Table 1.2 Supported standard and library name

Supported standard	Compiler	Library name
IEC62386-207 Edition 1.0	Renesas CC-RL V1.10.00	r_dali_207_cc_gen2_v1_00.lib
	IAR C/C++ Compiler for Renesas RL78 V4.21.2.2420	r_dali_207_iar_gen2_v1_00.a

1.4 File list

The list of files provided by this library is described below.

Table 1.3 File list

File name	Description
r_dali_207_cc_gen2_v1_00.lib r_dali_207_iar_gen2_v1_00.a	CC-RL version library file IAR version library file
r_dali207_api.h	Library header file
r_dali207_common.h	Definition header file used in multiple modules
r_dali207_var.h	Definition header file for variable modules

1.5 Resource

The library resources (ROM/RAM size and maximum stack size) required by this library are shown below. Table 1.4 Library resource(fixed) lists resources that do not depend on Control Gear implementation content, and Table 1.5 Library resource(variable) lists resources that do not depend on Control Gear implementation content.

Table 1.4 Library resource(fixed)

Compiler	Item		Size
CC-RL	Library resource	ROM size	2,281 [bytes]
		RAM size	0 [bytes]
	Maximum stack size		34 [bytes] (R_DALI207_InitLogicalUnit function)
IAR	Library resource	ROM size	2,835 [bytes]
		RAM size	0 [bytes]
	Maximum stack size		26 [bytes] (R_DALI207_InitLogicalUnit function)

Table 1.5 Library resource(variable)

Compiler	Item	Size
CC-RL	dali207_t	24 [bytes / logical unit]
IAR	dali207_t	24 [bytes / logical unit]

1.6 Development environment

The environment when developing this library is described below.

Table 1.6 Library development environment

Compiler	Item	Description
CC-RL	Integrated development environment	e2studio V2021-04
	C compiler	Renesas CC-RL V1.10.00
	CPU core	RL78-S2 core
	Optimization level	Code size precedence
	Language standard	GNU ISO C99
IAR	Integrated development environment	IAR Embedded Workbench for Renesas RL78 V8.5.2.7561
	C compiler	IAR C/C++ Compiler For Renesas RL78 V4.21.3.2447
	CPU core	RL78-S2 core
	Optimization level	Code size precedence
	Language standard	GNU ISO C99

1.7 Notes

1. The API functions in this library are prohibited from being called by the interrupt handler in the user application.
2. The loop processing of programs containing this library should be able to run for less than 1 ms at maximum. An environment in which loop processing runs for more than 1 ms will not meet the DALI standard specifications.
3. The dali207_t type structure is a reference-only structure.

2. Programming environment

The hardware and software environments required for users to perform Control Gear operations using this library are described in this section.
Only the requirements that are required in addition to those in the DALI102 library are described.

2.1 Hardware requirement

2.1.1 LED dimming control circuit

As a dimming control circuit, a control circuit capable of dimming LED lamps is required.

2.1.2 Failure detection mechanism

Control Gear needs to detect operational failures, retain the status in an internal variable, and then respond to the Application Controller's inquiry. For this reason, a hardware failure detection mechanism (e.g., lighting fixture failure) is required.

2.1.2.1 LED lamp circuit short detection

This is a short-circuit detection mechanism for the circuit between the LED lamp connected to the Control Gear.
This mechanism is optional.

2.1.2.2 LED lamp circuit open detection

This is an open detection mechanism for the circuit between the LED lamp and the LED lamp connected to the Control Gear.
This mechanism is optional.

2.1.2.3 LED lamp load measurement

This is a mechanism to measure the load (current) of the LED lamp.
This mechanism is optional.

2.1.2.4 Temperature measurement

This is a mechanism to measure the temperature of the device.
This mechanism is optional.

2.2 Software requirement

2.2.1 DALI207 Module Definition

A unit of a logical Bus unit (corresponding to Control Gear in this document) defined in one piece of hardware is called a logical unit. This library provides a structure type (dali207_t) that summarizes the parameters necessary to extend the DALI207 standard for the DALI102 module provided by the DALI102 library. A dali207_t type variable is called a DALI207 module.
Please define DALI207 modules for the number of logical units that need to be extended with DALI207.

2.2.2 Dimming process

In addition to the actual level obtained with the R_DALI102_GetActualLevel function of the DALI102 library, calculate the dimming rate corresponding to the actual level according to the dimming curve obtained with the R_DALI207_GetDimmingCurve function, and reflect it in the LED lamp.

2.2.3 LED lamp power measurement

Implement a function to measure the load power applied to LED lamps.
This feature is optional.

2.2.3.1 Reference power measurement

Call the R_DALI207_GetStatus function periodically to measure the load power applied to the lamp while the reference_measurement_active bit in the return value is true and save it as the reference power in the non-volatile memory.
This feature is optional.

2.2.3.2 Current protection

While the current_protector_active bit in the return value of the function is true, measure the power of the LED lamps periodically, and if the difference (⊿P) from the reference voltage measured by the reference power measurement function exceeds a certain level, turn off the output of the LED lamps.
The threshold for turning off the output of the LED lamps is user dependent. Set an appropriate value according to the hardware configuration and decide.
This feature is optional.

2.2.4 Failure detection/notification

When an error status occurs or is resolved in the implemented failure detection function, call the following API function. Also, when an error is detected, periodically check whether the error status has been resolved or not. Some failures may be categorized as Lamp Failure. If at least one of the failures classified as Lamp Failure has occurred, call the R_DALI102_SetLampFailure function in the DALI102 library, and if all the failures have been resolved, call the R_DALI102_ClearLampFailure function. The error status classified as "Failure" is described below.

Table 2.1 Failure Classification Table of Lamp Failure

Failures	Categorized as Lamp Failure
SHORT CIRCUIT	✓
OPEN CIRCUIT	✓
LOAD DECREASE	✓
LOAD INCREASE	✓
CURRENT PROTECTOR ACTIVE	✓
THERMAL OVERLOAD	-
THERMAL SHUT DOWN	-
REFERENCE MEASUREMENT FAILED	-

2.2.4.1 SHORT CIRCUIT

A short circuit is detected in the circuit of the LED lamp. In this state, the detected LED lamp cannot be turned on.
When this status occurs, set the DALI207_FAILURE_STATUS_SHORT_ CIRCUIT parameter in the R_DALI207_AddFailureStatus function.
This feature is optional.

2.2.4.2 OPEN CIRCUIT

An open circuit is detected in the circuit of the LED lamp. In this state, the detected LED lamp cannot be turned on.
When this status occurs, set the DALI207_FAILURE_STATUS_OPEN_ CIRCUIT parameter in the R_DALI207_AddFailureStatus function.
This feature is optional.

2.2.4.3 LOAD DECREASE

This refers to a state in which the load on the LED lamp is significantly reduced based on the power value measured by the reference power measurement. The threshold to determine this state should be set by the user.
When this status is reached, set the DALI207_FAILURE_STATUS_LOAD_ DECREASE parameter in the R_DALI207_AddFailureStatus function.
This feature is optional.

2.2.4.4 LOAD INCREASE

This refers to a state in which the load on the LED lamp is significantly increased based on the power value measured by the reference power measurement. The threshold to determine this state should be set by the user.
When this status is reached, set the DALI207_FAILURE_STATUS_LOAD_INCREASE parameter in the R_DALI207_AddFailureStatus function.
This feature is optional.

2.2.4.5 CURRENT PROTECTOR ACTIVE

This refers to the state in which the LED is turned off for safety reasons.
The condition for activation is that the current protection feature is enabled (current_protector_enabled is true in the return value of the R_DALI207_GetStatus function) and the load is increasing/decreasing more than a certain level based on the power value measured by the reference power measurement.
When this state is reached, set the DALI207_FAILURE_STATUS_ CURRENT_PROTECTOR_ACTIVE parameter in the R_DALI207_AddFailureStatus function.
Also, when the dimming rate should be set to 0%, it should be deactivated and set the DALI207_FAILURE_STATUS_CURRENT_PROTECTOR_ACTIVE parameter in the R_DALI207_RemoveFailureStatus function.
This feature is optional.

2.2.4.6 THERMAL OVERLOAD

This refers to a state in which the temperature exceeds the maximum allowable temperature. The allowable temperature depends on the hardware and software.
When this status is reached, set the DALI207_FAILURE_STATUS_ THERMAL_OVERLOAD parameter in the R_DALI207_AddFailureStatus function.
This feature is optional.

2.2.4.7 THERMAL SHUT DOWN

This refers to the state in which the THERMAL OVERLOAD state is continuously occurring, and the LED is turned off for safety reasons. The time it takes to reach this state is user dependent.
When this state is reached, use the R_DALI207_AddFailureStatus function to set DALI207_FAILURE_STATUS_
THERMAL_SHUT_DOWN parameter in the R_DALI207_AddFailureStatus function.
This feature is optional.

2.2.4.8 REFERENCE MEASUREMENT FAILED

This refers to the state in which the reference power measurement feature is active (reference_measurement_active is true in the return value of the R_DALI207_GetStatus function) and the reference power measurement has failed.
When this state is reached, set the R_DALI207_AddFailureStatus function to DALI207_FAILURE_STATUS_ REFERENCE_MEASUREMENT_FAILED parameter in the R_DALI207_AddFailureStatus function.
This feature is optional.

2.2.5 Operating Mode

In the implemented operation mode, call the following API function when there is a change in the active state.

2.2.5.1 PWM dimming mode

This is the mode for dimming LED lamps by PWM control.
When it becomes active, set the DALI207_OPERATING_MODE parameter with the R_DALI207_AddOperatingMode function.
When it becomes inactive, set the DALI207_OPERATING_MODES_PWM_MODE parameter in the R_DALI207_RemoveOperatingMode function.
This feature is optional.

2.2.5.2 AM dimming mode

This is the mode for dimming LED lamps by AM control.
When it becomes active, set the DALI207_OPERATING_AM_MODE parameter with the R_DALI207_AddOperatingMode function.
When it becomes inactive, set the DALI207_OPERATING_AM_MODE parameter in the R_DALI207_RemoveOperatingMode function.
This feature is optional.

2.2.5.3 Output current control mode

This is the mode to control the output current of LED lamps.
When it becomes active, set the DALI207_OPERATING_CURRENT_CONTROL_MODE parameter with the R_DALI207_AddOperatingMode function.
When it becomes inactive, set the DALI207_OPERATING_CURRENT_CONTROL_MODE parameter in the R_DALI207_RemoveOperatingMode function.
This feature is optional.

2.2.5.4 High current pulse mode

This is the mode to output high current pulses to the LED lamps.
When it becomes active, set the DALI207_OPERATING_ HIGH_CURRENT_PULSE_MODE parameter with the R_DALI207_AddOperatingMode function.
When it becomes inactive, set the DALI207_OPERATING_ HIGH_CURRENT_PULSE_MODE parameter in the R_DALI207_RemoveOperatingMode function.
This feature is optional.

3. DALI207 library feature

The features of this library are described below.

3.1 Definition of data types and return values

The data types provided by this library are described below.

Table 3.1 List of data types

Type	Description
dali207_t	DALI207 module type

The definition macros provided by this library are described below.

Table 3.2 List of failure status

Macro name	Macro value	Description
DALI207_FAILURE_STATUS _SHORT_CIRCUIT	0x01	LED circuit short state
DALI207_FAILURE_STATUS _OPEN_CIRCUIT	0x02	LED circuit open state
DALI207_FAILURE_STATUS _LOAD_DECREASE	0x04	LED load decrease state
DALI207_FAILURE_STATUS _LOAD_INCREASE	0x08	LED load increase state
DALI207_FAILURE_STATUS _CURRENT_PROTECTOR_ACTIVE	0x10	LED current protection state
DALI207_FAILURE_STATUS _THERMAL_SHUT_DOWN	0x20	LED turned off due to continuous maximum allowable temperature
DALI207_FAILURE_STATUS _THERMAL_OVERLOAD	0x40	Maximum allowable temperature exceeded state
DALI207_FAILURE_STATUS _REFERENCE_MEASUREMENT_FAILED	0x80	LED reference power measurement failure state
DALI207_FAILURE_STATUS_ALL	0xFF	Failure state: All

Table 3.3 List of dimming curve

Macro name	Macro value	Description
DALI207_DIMMING_CURVE_LOGARITHMIC	0	Logarithmic curve
DALI207_DIMMING_CURVE_LINEAR	1	linear curve

Table 3.4 List of gear type

Macro name	Macro value	Description
DALI207_GEAR_TYPE_LED_POWER_SUPPLY_INTEGRATED	0x01	Control gear and LED power supply are integrated
DALI207_GEAR_TYPE_LED_MODULE_INTEGRATED	0x02	LED module is integrated into control gear
DALI207_GEAR_TYPE_AC_SUPPLY_POSSIBLE	0x04	AC power supply available
DALI207_GEAR_TYPE_DC_SUPPLY_POSSIBLE	0x08	DC power supply available

Table 3.5 List of operating modes

Macro name	Macro value	Description
DALI207_OPERATING_MODES_PWM_MODE	0x01	PWM dimming mode
DALI207_OPERATING_MODES_AM_MODE	0x02	AM dimming mode
DALI207_OPERATING_MODES _CURRENT_CONTROL_MODE	0x04	Output current control mode
DALI207_OPERATING_MODES _HIGH_CURRENT_PULSE_MODE	0x08	High current pulse mode

Table 3.6 List of features

Macro name	Macro value	Description
DALI207_FEATURES _SHORT_CIRCUIT_DETECTION	0x01	LED circuit short detection feature
DALI207_FEATURES_OPEN_CIRCUIT_DETECTION	0x02	LED circuit open detection feature
DALI207_FEATURES _DETECTION_OF_LOAD_DECREASE	0x04	LED load decrease detection feature
DALI207_FEATURES _DETECTION_OF_LOAD_INCREASE	0x08	LED load increase detection feature
DALI207_FEATURES _CURRENT_PROTECTOR_IS_IMPLEMENTED	0x10	LED current protection
DALI207_FEATURES _THERMAL_SHUT_DOWN	0x20	Maximum allowable temperature exceeded detection feature
DALI207_FEATURES_LIGHT_LEVEL _REDUCTION_DUE_TO_OVER_TEMPERATURE	0x40	Feature to turn off the LED due to continuous maximum allowable temperature exceedance

The return values provided by this library are described below.

Table 3.7 List of return value(dali207_return_t)

Definition	Return value	Description
DALI207_RETURN_OK	0	Normal end
DALI207_RETURN_ERR	1	Error end

3.2 List of structures

The structures provided by this library are described below.

Definition of the status value type structure (dali207_default_t)

```
typdef struct
{
    bool reference_measurement_active : 1;
    bool current_protector_enabled : 1;
} dali207_status_t;
```

Definition of the default value type structure (dali207_default_t)

```
typdef struct
{
    uint8_t min_fast_fade_time;
    uint8_t gear_type;
    uint8_t possible_operating_modes;
    uint8_t features;
    uint8_t phm;
} dali207_default_t;
```

Definition of the NVM variable type structure (dali207_nvm_t)

```
typdef struct
{
    uint8_t fast_fade_time;
    uint8_t dimming_curve;
    bool reference_measurement_failed : 1;
    bool current_protector_enabled : 1;
} dali207_nvm_t;
```

3.3 List of API Functions

The API functions of this library are described below.

Table 3.8 List of API functions

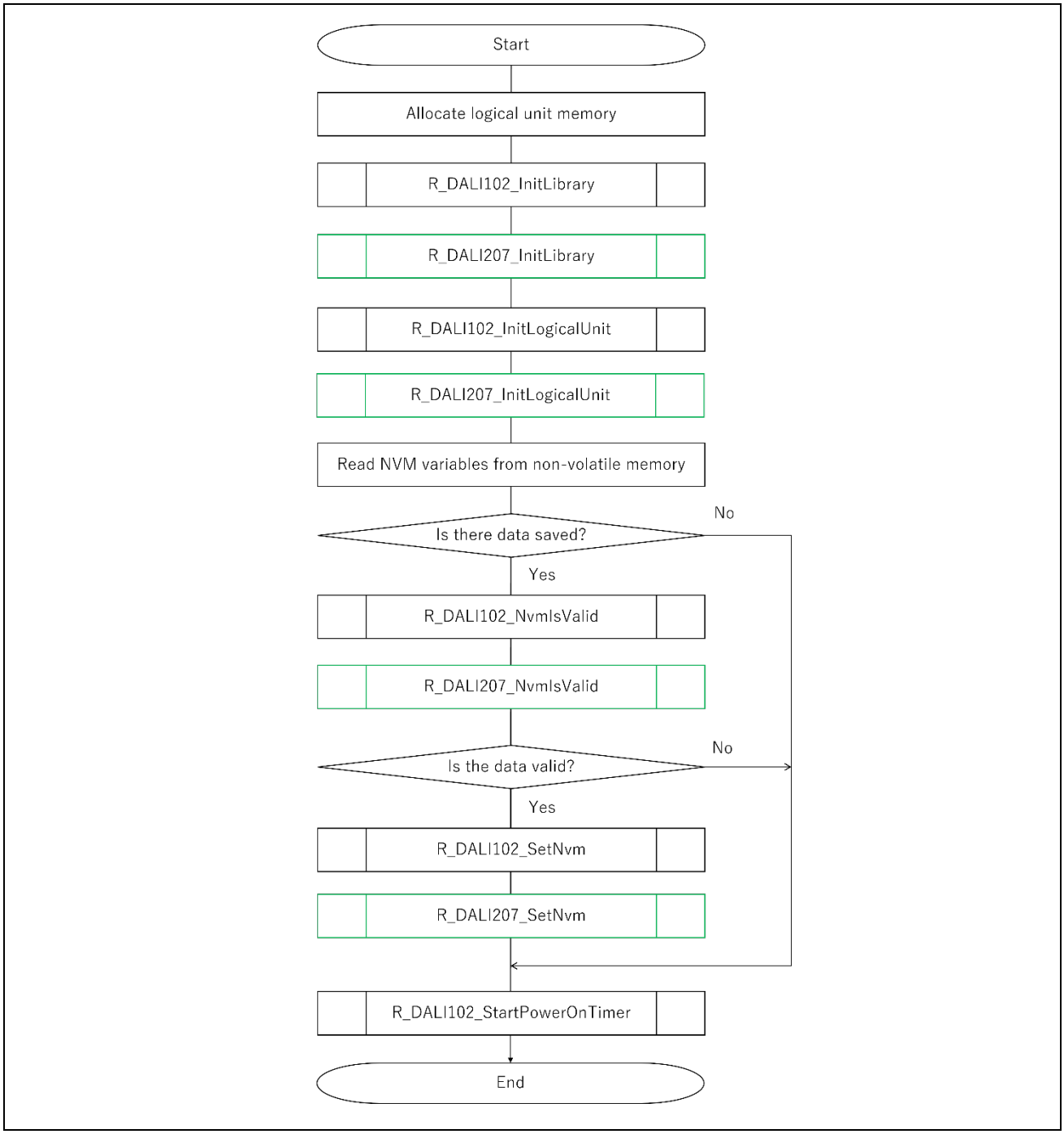
Function name	Description
R_DALI207_InitLibrary	Initialize the DALI207 library
R_DALI207_InitLogicalUnit	Initialize the logical unit
R_DALI207_SetNvm	Set the NVM variable value
R_DALI207_GetNvm	Get the NVM variable value
R_DALI207_NvmIsValid	Check NVM variable values within valid range
R_DALI207_NvmIsChanged	Check for NVM variable value change
R_DALI207_GetStatus	Get status
R_DALI207_GetDimmingCurve	Get dimming curve
R_DALI207_SetFailureStatus	Set failure status
R_DALI207_AddFailureStatus	Add failure status
R_DALI207_RemoveFailureStatus	Remove failure status
R_DALI207_SetOperatingMode	Set operating mode
R_DALI207_AddOperatingMode	Add operating mode
R_DALI207_RemoveOperatingMode	Remove operating mode
R_DALI207_FinishReferenceMeasurement	End reference measurement
R_DALI207_GetLibraryVersion	Get library version

3.4 Schematic flowchart

In addition to the environment for using the DALI102 library, the flow chart for using this library is described below.
The functions in the green boxes on the flowchart indicate the API functions of this library.

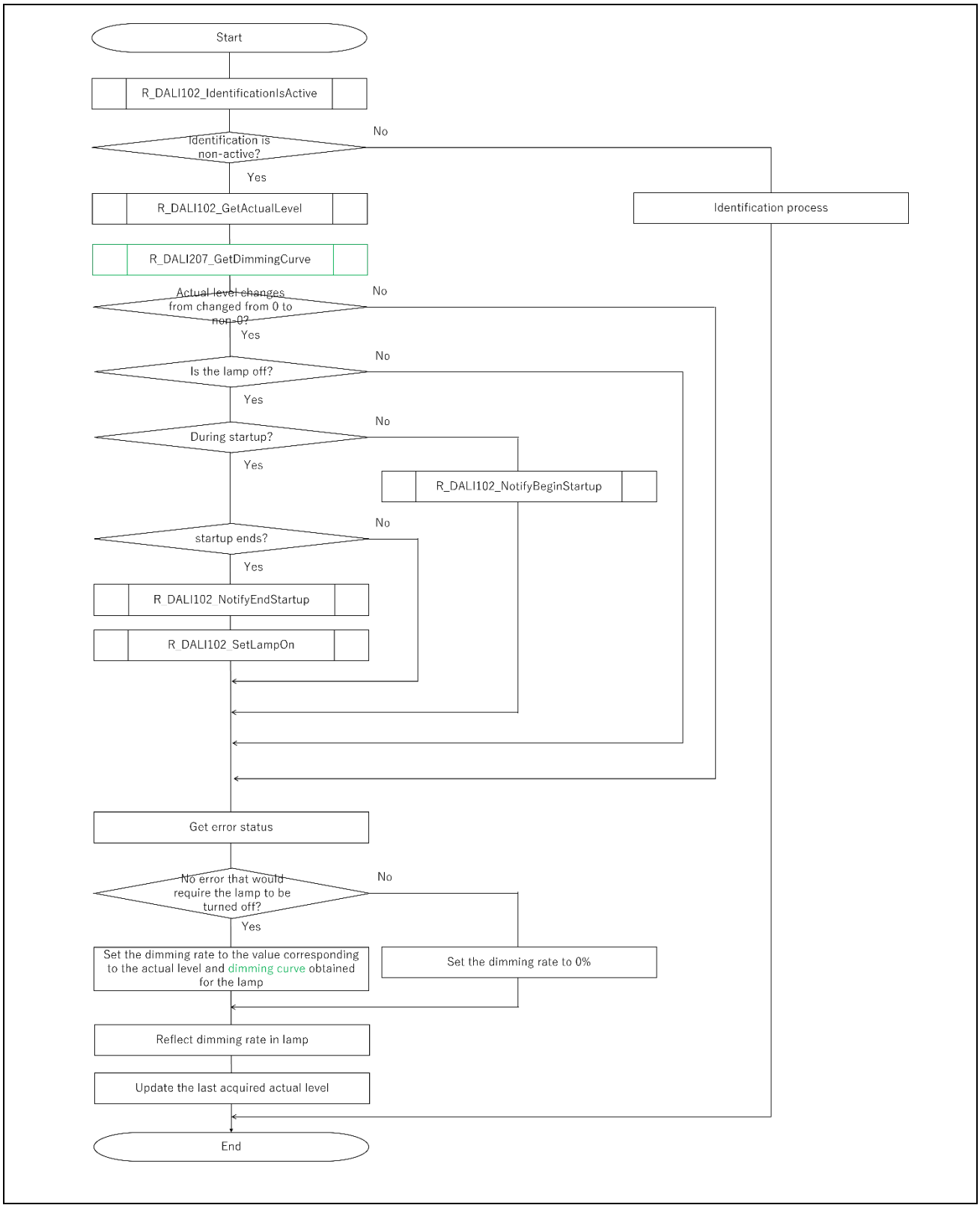
3.4.1 Initialization

The initialization flow is described below.



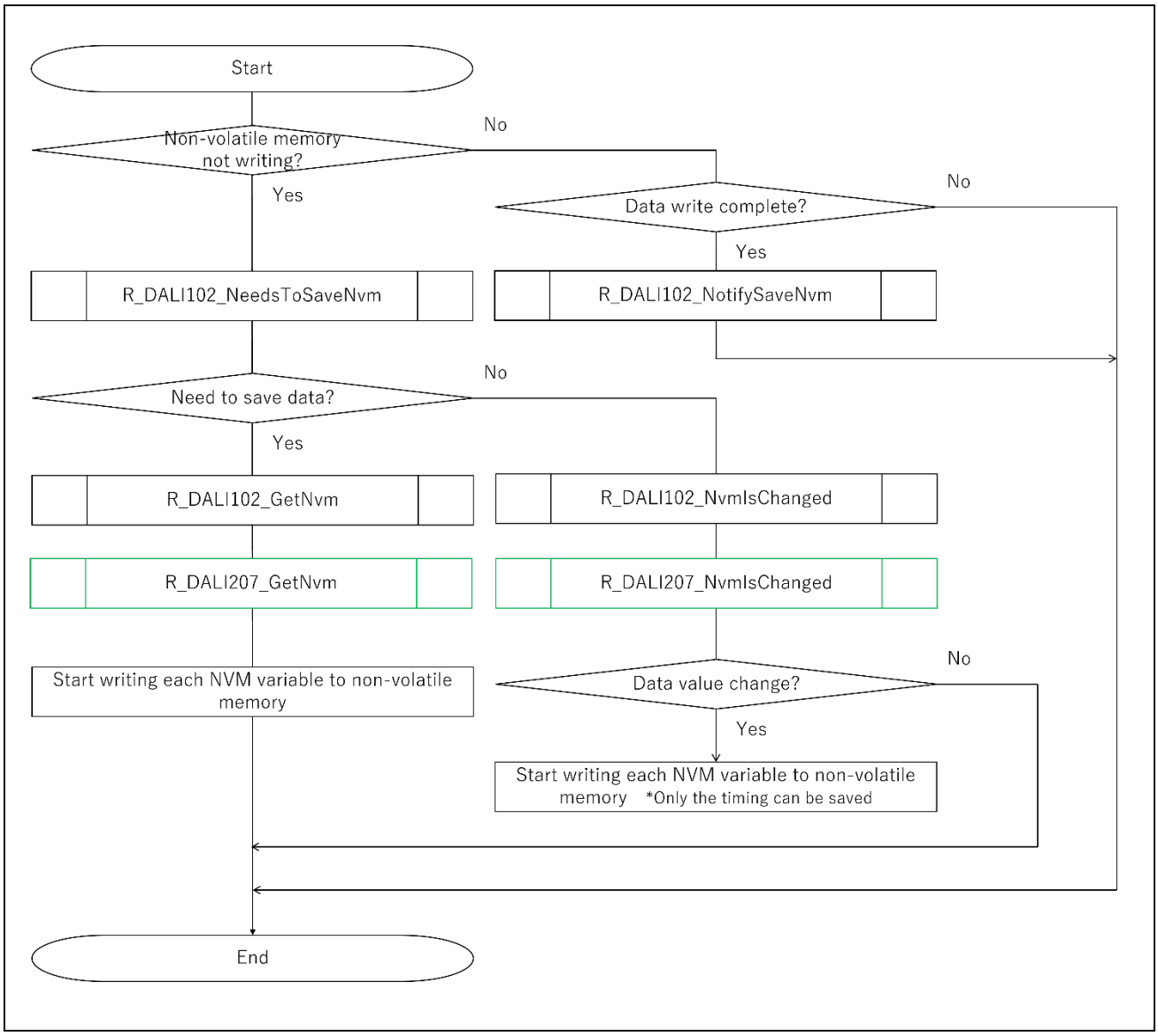
3.4.2 Dimming processing

This section describes the flow of the dimming process. Perform the process periodically.



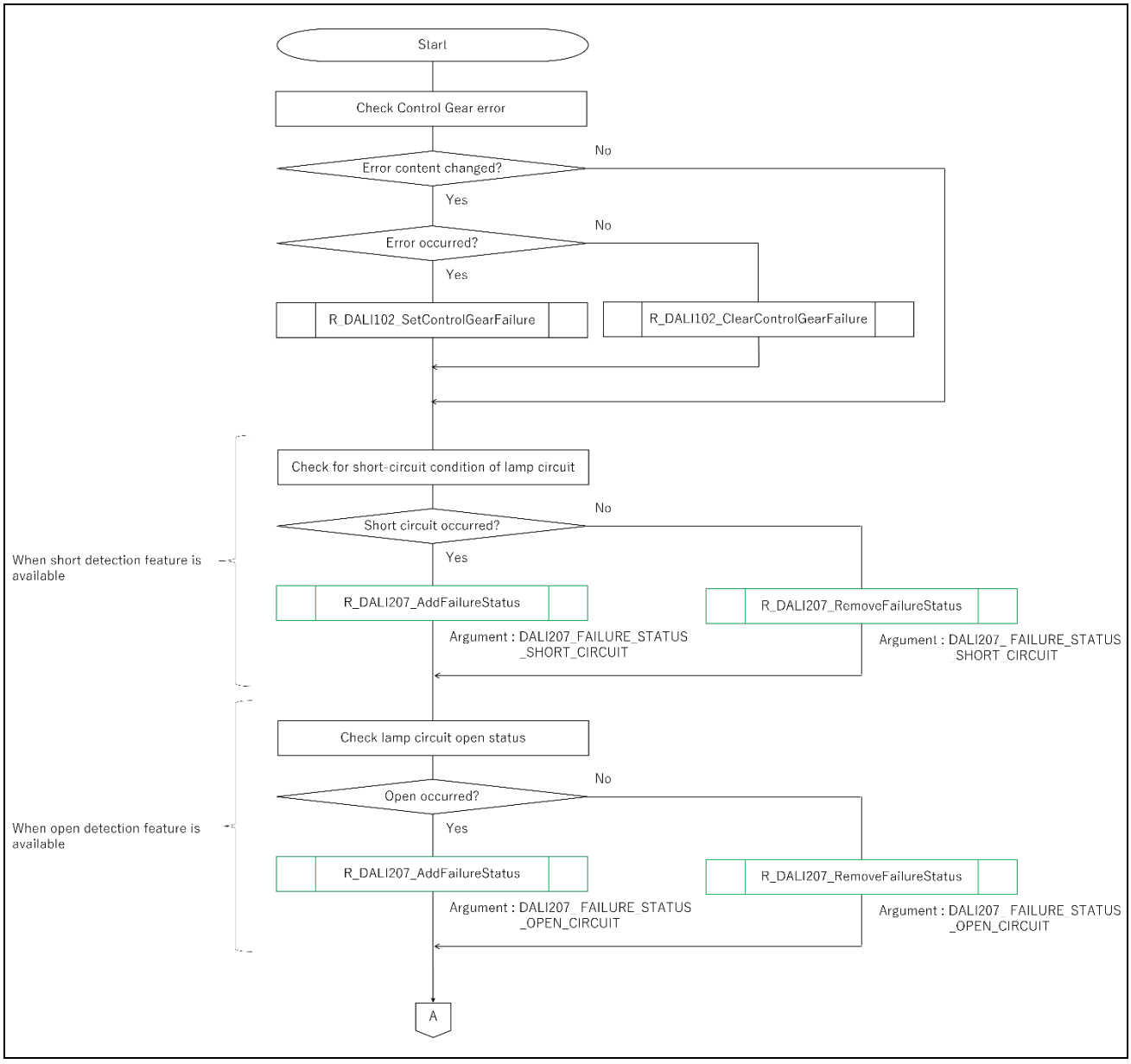
3.4.3 Non-volatile Data Processing

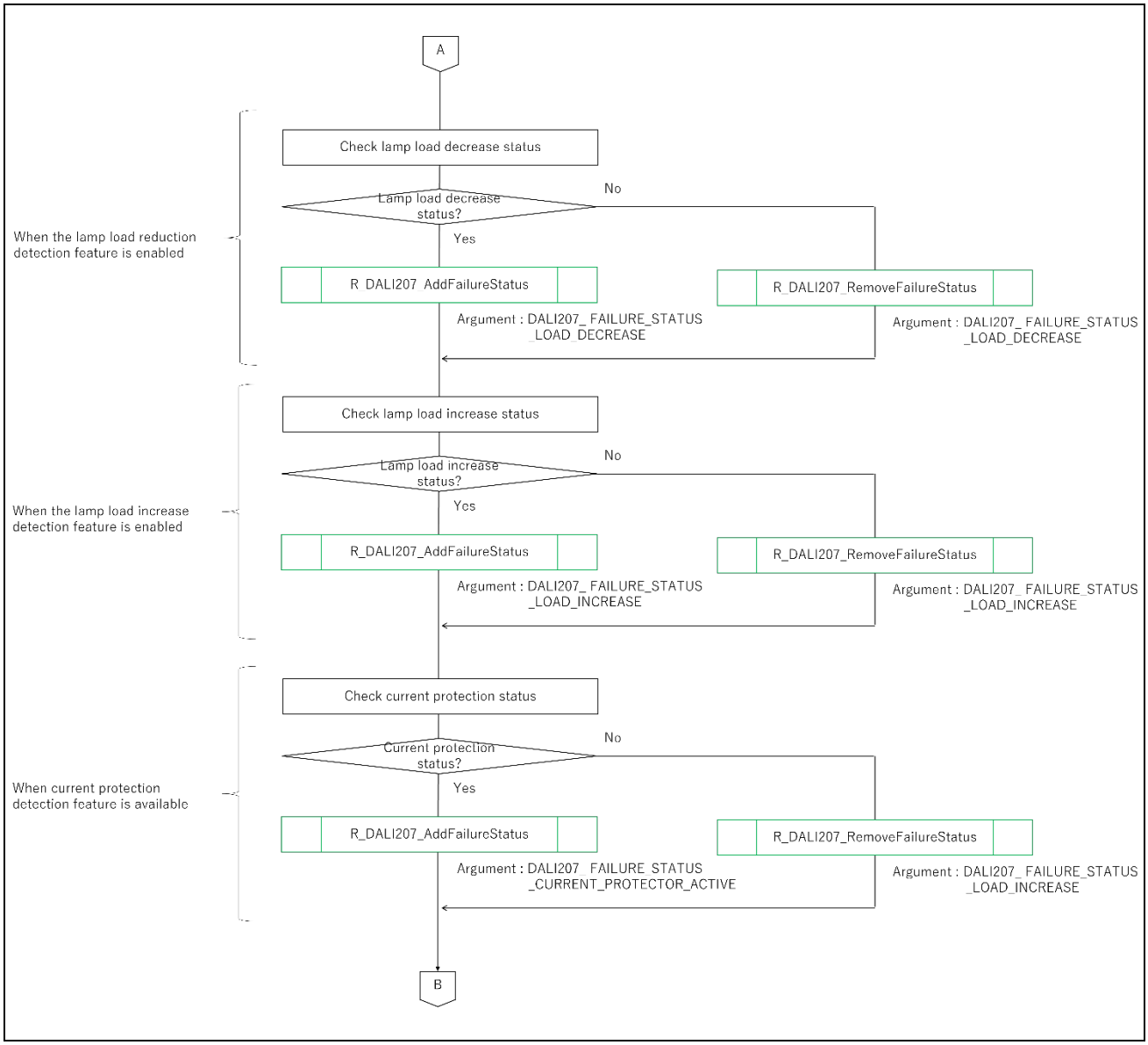
This section describes the flow of non-volatile data processing.

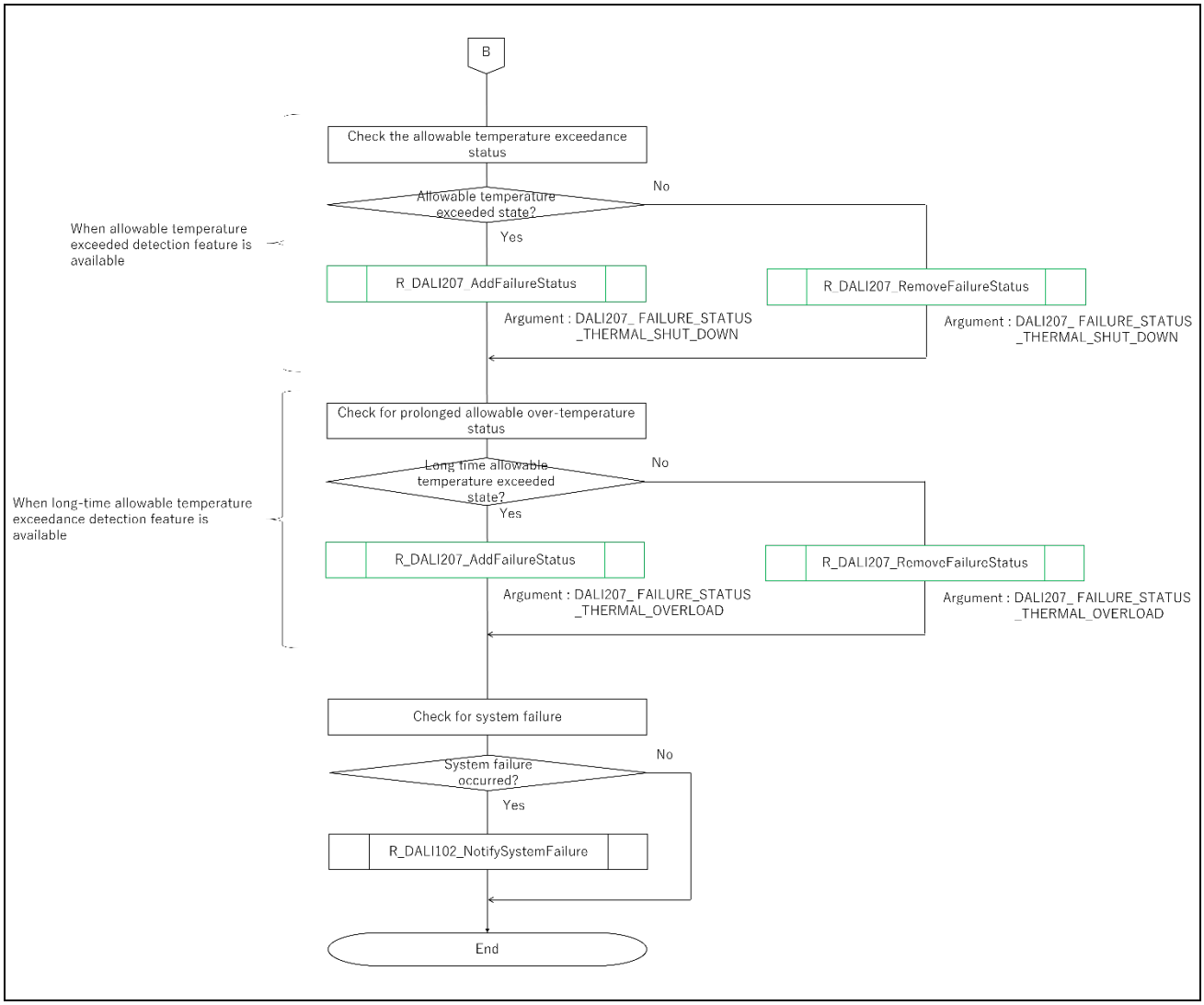


3.4.4 Error Handling

The flow of error handling is described below. Call this function when the error status is updated.
The detailed specifications of Control Gear error and Lamp error depend on the hardware and software.
Define the specifications according to the environment and consider implementation.

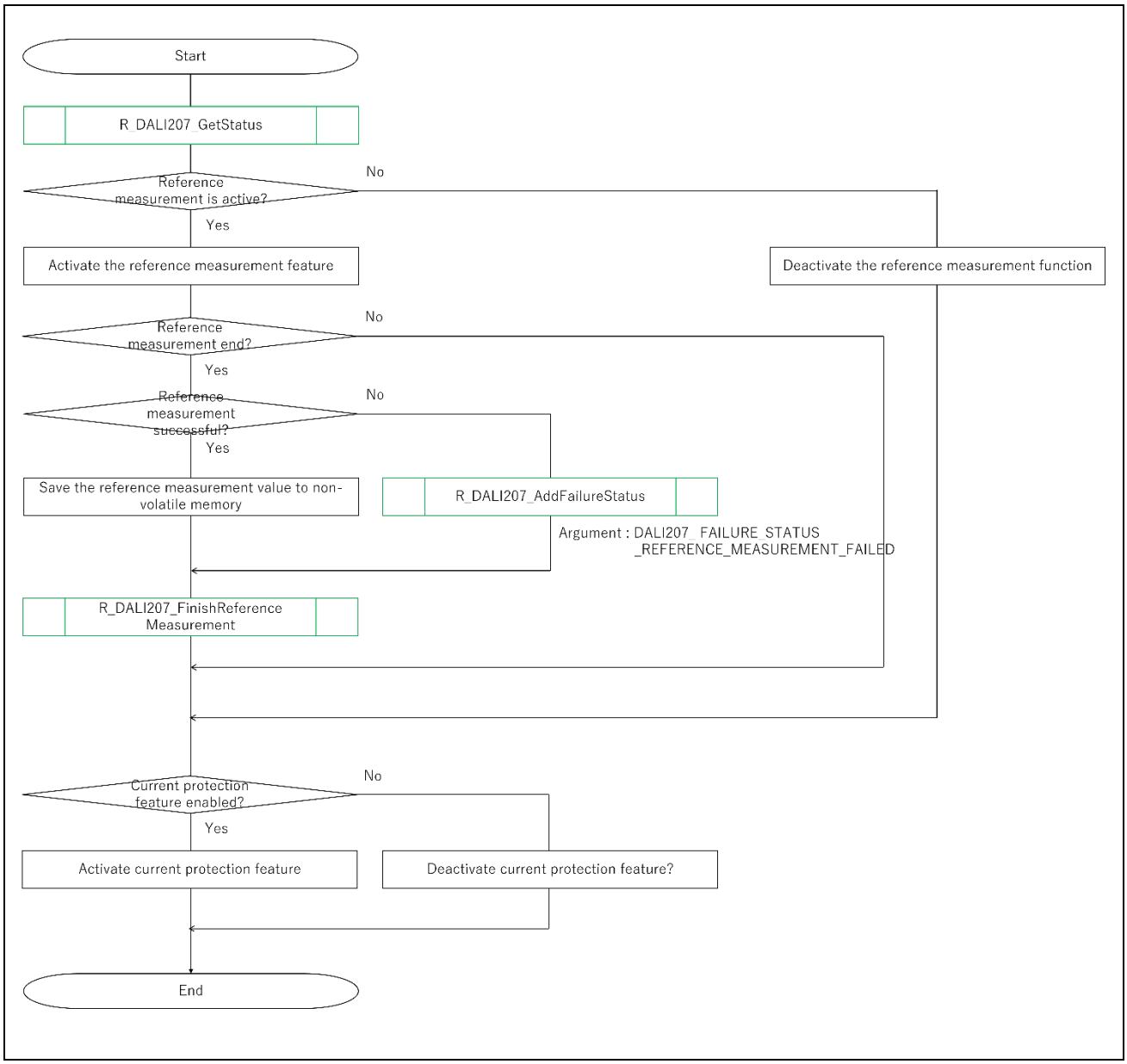



A
Check lamp load decrease status
Lamp load decrease status?
No
Yes
When the lamp load reduction detection feature is enabled
R_DALI207_AddFailureStatus
R_DALI207_RemoveFailureStatus
Argument : DALI207_FAILURE_STATUS _LOAD_DECREASE
Argument : DALI207_FAILURE_STATUS _LOAD_DECREASE
Check lamp load increase status
Lamp load increase status?
No
Yes
When the lamp load increase detection feature is enabled
R_DALI207_AddFailureStatus
R_DALI207_RemoveFailureStatus
Argument : DALI207_FAILURE_STATUS _LOAD_INCREASE
Argument : DALI207_FAILURE_STATUS _LOAD_INCREASE
Check current protection status
Current protection status?
No
Yes
When current protection detection feature is available
R_DALI207_AddFailureStatus
R_DALI207_RemoveFailureStatus
Argument : DALI207_FAILURE_STATUS _CURRENT_PROTECTOR_ACTIVE
Argument : DALI207_FAILURE_STATUS _LOAD_INCREASE
B


B
Check the allowable temperature exceedance status
Allowable temperature exceeded state?
No
Yes
When allowable temperature exceeded detection feature is available
R_DALI207_AddFailureStatus
R_DALI207_RemoveFailureStatus
Argument : DALI207_ FAILURE_STATUS _THERMAL_SHUT_DOWN
Argument : DALI207_ FAILURE_STATUS _THERMAL_SHUT_DOWN
Check for prolonged allowable over-temperature status
Long time allowable temperature exceeded state?
No
Yes
When long-time allowable temperature exceedance detection feature is available
R_DALI207_AddFailureStatus
R_DALI207_RemoveFailureStatus
Argument : DALI207_ FAILURE_STATUS _THERMAL_OVERLOAD
Argument : DALI207_ FAILURE_STATUS _THERMAL_OVERLOAD
Check for system failure
System failure occurred?
No
Yes
R_DALI102_NotifySystemFailure
End

3.4.5 Status Processing

This section describes the flow of status processing. Use the function according to the status condition.



3.5 API Function Specifications

The API function specifications for this library are listed below.

3.5.1.1 R_DALI207_InitLibrary

[Overview]
Initializes the DALI207 library.

[Format]

```
void R_DALI207_InitLibrary ( void )
```

[Prerequisite]
1. R_DALI102_InitLibrary function must have ended normally.

[Arguments]
None

[Return values]
None

3.5.1.2 R_DALI207_InitLogicalUnit

[Overview]

Initializes the specified logical unit.

Specify and call the DALI102 module of the logical unit to be extended by DALI207.

[Format]

```
dali207_return_t R_DALI207_InitLogicalUnit ( dali207_t * p_this,
                                             dali102_t * p_dali102,
                                             const dali207_default_t * p_default_value )
```

[Prerequisite]

1. R_DALI102_InitLibrary function must have ended normally.
2. R_DALI207_InitLibrary function must have ended normally.
3. R_DALI102_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
dali207_t * p_this	Pointer to DALI207 module
dali102_t * p_dali102	Pointer to DALI102 module
const dali207_default_t * p_default_value	factory burn-in default value Valid range: - min_fast_fade_time: 1 to 27 - gear_type: Refer to the next section for the valid range. - possible_operating_modes: Refer to the next section for the valid range. - features: Refer to the next section for the valid range. - phm: 1 to 254 * PHM value for a straight curve

[Return values]

Value	Description
DALI207_RETURN_OK	Normal end
DALI207_RETURN_ERR	Parameter error - Review the argument settings.

(1) Set the gear_type parameter

Set the parameter to the one that meets the hardware specification of the Control Gear.
If it supports more than one, the parameter should be a logical OR of the ones that satisfy the specification.

Parameter	Content
DALI207_GEAR_TYPE_LED_POWER_SUPPLY_INTEGRATED	Control gear and LED power supply are integrated
DALI207_GEAR_TYPE_LED_MODULE_INTEGRATED	LED module is integrated into control gear
DALI207_GEAR_TYPE_AC_SUPPLY_POSSIBLE	AC power supply available
DALI207_GEAR_TYPE_DC_SUPPLY_POSSIBLE	DC power supply available

(2) Set the possible_operating_modes

Set the parameter to the mode that can be operated by the specified logical unit.
If it supports more than one, the parameter should be a logical OR of the ones that satisfy the specification.

Parameter	Content
DALI207_OPERATING_MODES_PWM_MODE	PWM dimming mode
DALI207_OPERATING_MODES_AM_MODE	AM dimming mode
DALI207_OPERATING_MODES_CURRENT_CONTROL_MODE	Output current control mode
DALI207_OPERATING_MODES_HIGH_CURRENT_PULSE_MODE	High current pulse mode

(3) Set the features

Set the features implemented by the specified logical unit in the parameter.
If it supports more than one, the parameter should be a logical OR of the ones that satisfy the specification.

Parameter	Content
DALI207_FEATURES_SHORT_CIRCUIT_DETECTION	LED circuit short detection feature
DALI207_FEATURES_OPEN_CIRCUIT_DETECTION	LED circuit open detection feature
DALI207_FEATURES_DETECTION_OF_LOAD_DECREASE	LED load decrease detection feature
DALI207_FEATURES_DETECTION_OF_LOAD_INCRASE	LED load increase detection feature
DALI207_FEATURES_CURRENT_PROTECTOR_IS_IMPLEMENT	LED current protection
DALI207_FEATURES_THERMAL_SHUT_DOWN	Maximum allowable temperature exceeded detection feature
DALI207_FEATURES_LIGHT_LEVEL_REDUCTION _DUE_TO_OVER_TEMPERATURE	Feature to turn off the LED due to continuous maximum allowable temperature exceedance

3.5.1.3 R_DALI207_SetNvm

[Overview]

Sets the NVM variable value in the DALI207 module.
Use this function to set the read data when the NVM variable data is stored in the non-volatile memory at power-on.

[Format]

```
void R_DALI207_SetNvm ( dali207_t * p_this,
                        const dali207_nvm_t * p_nvm )
```

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.
3. Make sure that the NVM variable is within the valid range in the R_DALI207_NvmIsValid function.

[Arguments]

Argument	Description
dali207_t * p_this	Pointer to DALI207 module
const dali207_nvm_t * p_nvm	Pointer to NVM variable for DALI207 module

[Return values]

None

3.5.1.4 R_DALI207_GetNvm

[Overview]

Gets the value of the NVM variable setting from the DALI207 module.
Use this function to store the latest NVM variable values in non-volatile memory.

[Format]

```
void R_DALI207_GetNvm (const dali207_t * p_this,
                       dali207_nvm_t * p_nvm )
```

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
const dali207_t * p_this	Pointer to DALI207 module
dali207_nvm_t * p_nvm	Pointer to NVM variable for DALI207 module

[Return values]

None

3.5.1.5 R_DALI207_NvmIsValid

[Overview]

Returns whether or not all the values set in the members of the dali207_nvm_t type variable are within the valid range.

Be sure to call the aforementioned R_DALI207_SetNvm function to check before setting the values.

[Format]

```
bool R_DALI207_NvmIsValid ( const dali207_t * p_this,
                            const dali207_nvm_t * p_nvm )
```

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
const dali207_t * p_this	Pointer to DALI207 module
const dali207_nvm_t * p_nvm	Pointer to NVM variable for DALI207 module Valid range: - fast_fade_time: 0x00, min_fast_fade_time to 27 - dimming_curve: 0x00, 0x01 - reference_measurement_failed: true, false - current_protector_enabled: true, false

[Return values]

Value	Description
true	All variables are within the valid range
false	At least one variable is outside the valid range

3.5.1.6 R_DALI207_NvmIsChanged

[Overview]

Gets whether there has been a change in at least one NVM variable value.
If the return value of this function is true, save the NVM variable to the non-volatile memory according to the hardware status.
The state that can be acquired by this function is the target from the time when this function was called last time (when it was called for the first time, when it was started).

[Format]

bool R_DALI207_NvmIsChanged (dali207_t * p_this)

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
dali207_t * p_this	Pointer to DALI207 module

[Return values]

Value	Description
true	Value change
false	No value change

3.5.1.7 R_DALI207_GetStatus

[Overview]

Gets the status. Call this function periodically to perform processing according to the status.

- While reference_measurement_active is true, perform the following actions.
  Perform the reference power measurement of the LED lamp.
  If the reference power measurement is successful, save the measured value to the non-volatile memory and call the R_DALI207_FinishReferenceMeasurement function to complete the reference power measurement.
  If the measurement fails, use the R_DALI207_AddFailureStatus function to specify the DALI207_FAILURE_STATUS_REFERENCE_MEASUREMENT_FAILED in the R_DALI207_AddFailureStatus function, and then call the R_DALI207_FinishReferenceMeasurement function to complete the reference power measurement.
- While current_protector_enabled is true, perform the following actions.
  Enable the current protection feature.

[Format]

dali207_status_t R_DALI207_GetStatus (const dali207_t * p_this)

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
const dali207_t * p_this	Pointer to DALI207 module

[Return values]

Bit	Value	Description
reference_measurement_active	true	Reference power measurement active of LED lamps
	false	Reference power measurement inactive for LED lamps
current_protector_enabled	true	Current protection feature enabled
	false	Current protection feature disabled

3.5.1.8 R_DALI207_GetDimmingCurve

[Overview]

Gets the dimming curve value during operation.
Call this function together with the R_DALI102_GetActualLevel function, which is obtained from the DALI102 library, and set the dimming curve to apply the actual level according to the obtained dimming curve and reflect the obtained dimming rate to the LED lamp.

[Format]

uint8_t R_DALI207_GetDimmingCurve (const dali207_t * p_this)

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
const dali207_t * p_this	Pointer to DALI207 module

[Return values]

Value	Description
0x00	Logarithmic curve $Dimming\ ratio[\%] = 10^{\frac{actualLevel-1}{253/3}-1}$
0x01	Linear curve $Dimming\ ratio[\%] = \frac{actualLeve}{254} * 100$

3.5.1.9 R_DALI207_SetFailureStatus

[Overview]

Sets the Failure Status at once. Call this function when the failure status is updated.
The parameter indicating the failure status can be set by logical OR.

[Format]

```
void R_DALI207_SetFailureStatus ( dali207_t * p_this, uint8_t status )
```

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
dali207_t * p_this	Pointer to DALI207 module
uint8_t status	Failure Status DALI207_FAILURE_STATUS_SHORT_CIRCUIT: LED lamp in short circuit state DALI207_FAILURE_STATUS_OPEN_CIRCUIT: LED lamp in open circuit state DALI207_FAILURE_STATUS_LOAD_DECREASE: LED lamp in load decrease state DALI207_FAILURE_STATUS_LOAD_INCREASE: LED lamp in load increase state DALI207_FAILURE_STATUS_CURRENT_PROTECTOR_ACTIVE: Current protection active state DALI207_FAILURE_STATUS_THERMAL_SHUT_DOWN: LED turned off during continuous maximum allowable DALI207_FAILURE_STATUS_THERMAL_OVERLOAD: Maximum allowable temperature exceeded state DALI207_FAILURE_STATUS_REFERENCE_MEASUREMENT_FAILED: LED reference power measurement failure state

[Return values]

None

3.5.1.10 R_DALI207_AddFailureStatus

[Overview]

Adds a specific state to Failure Status. Call this function when the failure status is updated.
The parameter indicating the failure status can be set by logical OR.

[Format]

```
void R_DALI207_AddFailureStatus ( dali207_t * p_this, uint8_t status )
```

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
dali207_t * p_this	Pointer to DALI207 module
uint8_t status	Failure Status DALI207_FAILURE_STATUS_SHORT_CIRCUIT: LED lamp in short circuit state DALI207_FAILURE_STATUS_OPEN_CIRCUIT: LED lamp in open circuit state DALI207_FAILURE_STATUS_LOAD_DECREASE: LED lamp in load decrease state DALI207_FAILURE_STATUS_LOAD_INCREASE: LED lamp in load increase state DALI207_FAILURE_STATUS_CURRENT_PROTECTOR_ACTIVE: Current protection active state DALI207_FAILURE_STATUS_THERMAL_SHUT_DOWN: LED turned off during continuous maximum allowable DALI207_FAILURE_STATUS_THERMAL_OVERLOAD: Maximum allowable temperature exceeded state DALI207_FAILURE_STATUS_REFERENCE_MEASUREMENT_FAILED: LED reference power measurement failure state

[Return values]

None

3.5.1.11 R_DALI207_RemoveFailureStatus

[Overview]

Removes a specific state from Failure Status. Call this function when the failure status is updated.
The parameter indicating the failure status can be set by logical OR.

[Format]

```
void R_DALI207_RemoveFailureStatus ( dali207_t * p_this, uint8_t status )
```

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
dali207_t * p_this	Pointer to DALI207 module
uint8_t status	Failure Status DALI207_FAILURE_STATUS_SHORT_CIRCUIT: LED lamp in short circuit state DALI207_FAILURE_STATUS_OPEN_CIRCUIT: LED lamp in open circuit state DALI207_FAILURE_STATUS_LOAD_DECREASE: LED lamp in load decrease state DALI207_FAILURE_STATUS_LOAD_INCREASE: LED lamp in load increase state DALI207_FAILURE_STATUS_CURRENT_PROTECTOR_ACTIVE: Current protection active state DALI207_FAILURE_STATUS_THERMAL_SHUT_DOWN: LED turned off during continuous maximum allowable DALI207_FAILURE_STATUS_THERMAL_OVERLOAD: Maximum allowable temperature exceeded state DALI207_FAILURE_STATUS_REFERENCE_MEASUREMENT_FAILED: LED reference power measurement failure state DALI207_FAILURE_STATUS_ALL: All failure states

[Return values]

None

3.5.1.12 R_DALI207_SetOperatingMode

[Overview]

Sets the active operation mode at once. Call this function when the active operation mode is updated.
The parameter indicating the operation mode can be set by logical OR.

[Format]

```
void R_DALI207_SetOperatingMode (dali207_t * p_this, uint8_t mode )
```

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
dali207_t * p_this	Pointer to DALI207 module
uint8_t mode	Operation mode DALI207_OPERATING_MODES_PWM_MODE: PWM dimming mode DALI207_OPERATING_MODES_AM_MODE: AM dimming mode DALI207_OPERATING_MODES_CURRENT_CONTROL_MODE: Output current control mode DALI207_OPERATING_MODES_HIGH_CURRENT_PULSE_MODE: High current pulse mode

[Return values]

None

3.5.1.13 R_DALI207_AddOperatingMode

[Overview]

Adds an active operating mode. Call this function when the active operation mode is updated.
The parameter indicating the operation mode can be set by logical OR.

[Format]

```
void R_DALI207_AddOperatingMode ( dali207_t * p_this, uint8_t mode )
```

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
dali207_t * p_this	Pointer to DALI207 module
uint8_t mode	Operation mode DALI207_OPERATING_MODES_PWM_MODE: PWM dimming mode DALI207_OPERATING_MODES_AM_MODE: AM dimming mode DALI207_OPERATING_MODES_CURRENT_CONTROL_MODE: Output current control mode DALI207_OPERATING_MODES_HIGH_CURRENT_PULSE_MODE: High current pulse mode

[Return values]

None

3.5.1.14 R_DALI207_RemoveOperatingMode

[Overview]

Removes the active operating mode. Call this function when the active operation mode is updated.
The parameter indicating the operation mode can be set by logical OR.

[Format]

```
void R_DALI207_RemoveOperatingMode ( dali207_t * p_this, uint8_t mode )
```

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
dali207_t * p_this	Pointer to DALI207 module
uint8_t mode	Operation mode DALI207_OPERATING_MODES_PWM_MODE: PWM dimming mode DALI207_OPERATING_MODES_AM_MODE: AM dimming mode DALI207_OPERATING_MODES_CURRENT_CONTROL_MODE: Output current control mode DALI207_OPERATING_MODES_HIGH_CURRENT_PULSE_MODE: High current pulse mode

[Return values]

None

3.5.1.15 R_DALI207_FinishReferenceMeasurement

[Overview]

While the return value reference_measurement_active of the R_DALI207_GetStatus function is true, call it when the reference power measurement is finished.

[Format]

void R_DALI207_FinishReferenceMeasurement (dali207_t * p_this)

[Prerequisite]

1. R_DALI207_InitLibrary function must have ended normally.
2. R_DALI207_InitLogicalUnit function must have ended normally.

[Arguments]

Argument	Description
dali207_t * p_this	Pointer to DALI207 module

[Return values]

None

3.5.1.16 R_DALI207_GetLibraryVersion

[Overview]

Gets the version number of this library.

[Format]

```
uint16_t R_DALI207_GetLibraryVersion ( void )
```

[Prerequisite]

None

[Arguments]

None

[Return values]

Value	Description
uint16_t	Version number (format: 0xXXYY) XX: Major version YY: Minor version

Revision History	RL78 Family Control Gear Library User's Manual: LED (207)

Rev.	Date	Description	
		Page	Summary
1.00	15th, Jun., 22	—	First Edition issued
1.01	1st, Nov., 22	—	Correct wrong description, Changed resource expression.

RL78 Family Control Gear Library
User's Manual: LED (207)

Publication Date: Rev.1.01 1st, Nov., 22

Published by: Renesas Electronics Corporation

RL78 Family

Renesas Electronics Corporation

R01US0536EJ0101