LoRaWAN® Stack

Sample Application

Introduction

This document describes a sample software to use LoRaWAN® stack. This application operates the LoRaWAN stack by user with some commands from a Host PC.

Notes on release

Features described with gray background are currently unsupported.

Target Devices

- MCU: Renesas RL78/G23 (R7F100GSN, R7F100GLG), RL78/G14 (R5F104ML), RL78/G22 (R7F102GGE), RA2E1 (R7FA2E1A9xxFM) or RA2L1 (R7FA2L1AB2DFP)
- Transceiver: Semtech SX1261 or SX1262

Note: When using RL78/G22, there are some limitations of LoRaWAN functions. See chapter 6.

Contents

1. Overview

This document describes the APIs for operating the LoRaWAN stack sample application program. This application can be operated by the AT command sets.

1.1 Environment

Figure 1 indicates the environment to use this sample application. This sample application can be operated with the AT command sets from a Host PC via a serial interface.

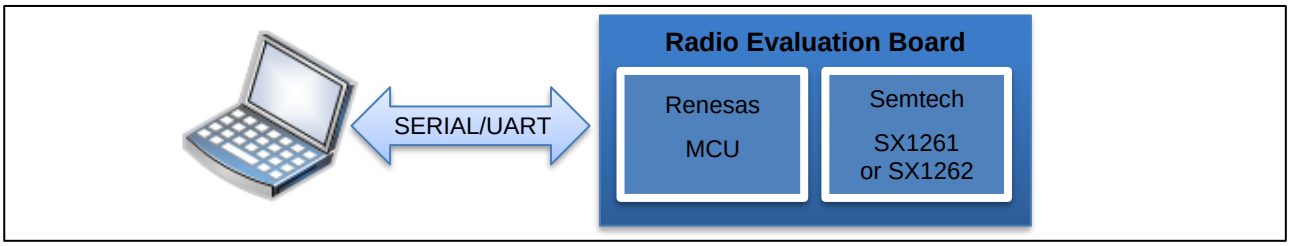

Figure 1. Sample Application Environment

The target MCU and the target RF for this sample application are Renesas MCU and Semtech SX1261 or SX1262.

1.2 Software Diagram

Figure 2 indicates a block diagram of this sample application.

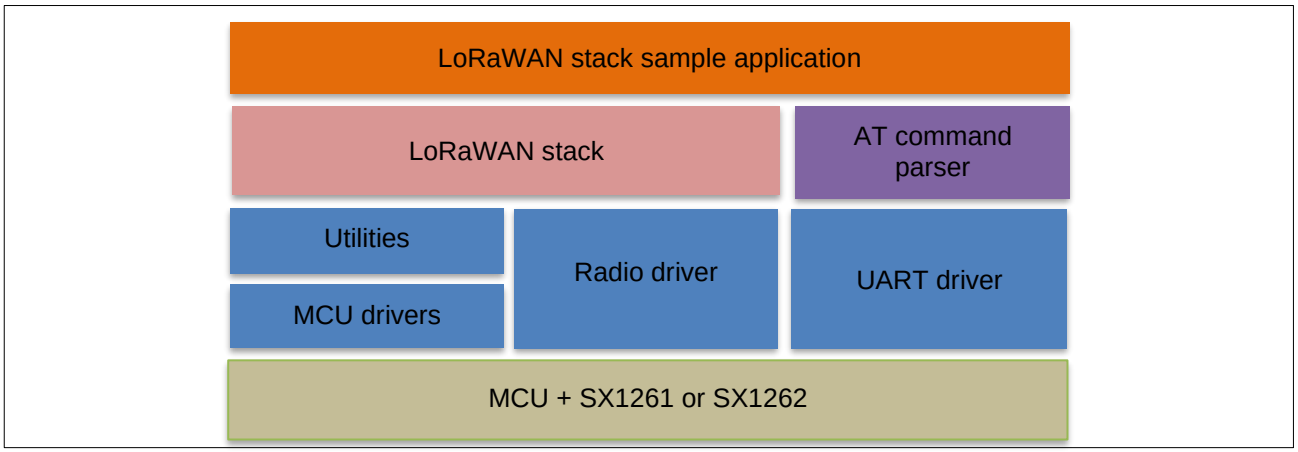

Figure 2. Sample Application Block Diagram

1.3 Directories (informative)

Table 1 shows a basic concept of what kind of codes each directory includes. This is just for information.

Table 1. Directories

Directories	Description
apps/LoRaSample	LoRaWAN sample application codes
boards	Board specific codes
boards/mcu	MCU drivers
mac	LoRaWAN MAC stack
radio	Radio driver for LoRa®
peripherals	Security related codes
system	Utility codes
system/flash/non-rfd/c_flash/FSL (*1)	Code Flash Library for RL78/G14 (*1)
system/flash/non-rfd/d_flash/EEPROMEMU (*2)	EEPROM Emulation Library for RL78/G14 (*2)
system/flash/rfd/rfd	Renesas Flash Driver RL78 Type01 for RL78/G23 (*3)
system/flash/	Flash drivers for RA2E1

CAUTION:

*1. This folder is for containing the code flash library to write to the code flash memory. The library for testing, 'Flash Self Programming Library Type01 Package Ver.3.00 for the RL78 Family [for the CA78K0R/CC-RL Compiler], is contained in advance. But when you start to develop a product, it is necessary to copy the newest code flash library to this folder.
The code flash library is downloaded from Renesas website.

*2. This folder is for containing the EEPROM emulation library and data flash access library to access the data flash memory. The libraries for testing, 'EEPROM Emulation Library Pack02 Package Ver.2.00(for CA78K0R/CC-RL Compiler) for RL78 Family', are contained in advance. But when you start to develop a product, it is necessary to copy the newest library to this folder.
The EEPROM emulation library and data flash access library are downloaded from Renesas website.

*3. This folder is for containing the flash driver to access the data and code flash memory. The driver for testing, 'Renesas Flash Driver RL78 Type 01 V1.00 for RL78/G23', is contained in advance. But when you start to develop a product, it is necessary to copy the newest library to this folder.
The flash driver is downloaded from Renesas website.

1.4 Resource Usage

Please refer to [7] for RL78 and [8] for RA in the following folder as for the resource usage such as memory and peripherals.

Folder: (package top)\documents\

1.5 Serial Terminal Software Configuration

The Serial Terminal Software configuration of this sample application is as shown in Table 2.

Table 2. Serial Terminal Software Configuration

Configuration Items	Value
Baud rate	115200 bps
Data bit	8 bits
Parity bit	None
Stop bit	1 bit
Flow control	None
Local echo back	Yes
Line terminator	Transmission: CR+LF Reception: CR+LF

1.6 Related Documentation

Table 3. Related Documentation

	Document No.	Title	Author	Language
[1]	R11AN0228	LoRaWAN® Stack Reference Guide	Renesas Electronics	English
[2]	R11AN0227	Radio Driver Reference Guide	Renesas Electronics	English
[3]	R11AN0834	Radio Driver Support Functions for Regional Radio Regulations	Renesas Electronics	English
[4]	R30UZ0095	Renesas LPWA Studio	Renesas Electronics	English
[5]	MCP-AA-22-0057-2	RL78 LoRaWAN® Sensor Demo Tutorial Setup and Operation method	Renesas Electronics	English
[6]	MCP-AA-22-0133-1	RA LoRaWAN® Sensor Demo Tutorial Setup and Operation Method	Renesas Electronics	English
[7]	R11AN0595	RL78/G23, RL78/G22, RL78/G14 LoRa®-based Wireless Software Package	Renesas Electronics	English
[8]	R11AN0596	RA2E1, RA2L1 LoRa®-based Wireless Software Package	Renesas Electronics	English

1.7 Restrictions

Table 4. Restrictions

Item	Restriction
Certification test feature	Supports the regions under the conditions as follows. • In case of LoRaWAN v1.0.4, Class A and OTAA Region AS923(Group 1 to 4), AU915, EU868, IN865, KR920 and US915 • In case of LoRaWAN v1.0.3 which encompasses v1.0.2, Class A and OTAA Region AS923, EU868, IN865, KR920 and US915

2. AT Command Syntax

2.1 Command Syntax

A command line is made up of three elements: the prefix, the body, and the termination character. The body is made up of individual commands as specified later in this document.

Note: This sample application program supports up to 512 characters for input of the command line. If the number of input characters exceeds the limit, the exceeded characters are ignored except for <BS>.

Table 5. General Command Syntax

Prefix	Body	Termination
“AT”	Individual commands as specified	<CR><LF>

2.1.1 Basic Command Syntax

The format of basic command syntax consists of <command>[<number>], where <command> is a single alphabetic character and where <number> may have a string of one or more characters from “0” to “9” decimal integer value.

Table 6. Basic Command Syntax

Prefix	Body	Termination	Description
“AT”	<command>[<number>]	<CR><LF>	Execute a command with number

2.1.2 Extended Command Syntax

There are two types of commands: action commands and parameter commands.

Action commands are used to “execute” a particular function of the equipment. Parameter commands are used to “set” value(s) to parameter(s) or to “read” current value(s) of parameter(s).

Table 7. Extended Command Syntax: Action Command

Prefix	Body (Subcommand)	Termination	Description
“AT”	+<name>	<CR><LF>	Execute an action command with no parameters
“AT”	+<name>=<value>|<compound values>	<CR><LF>	Execute an action command with one or more parameters

Table 8. Extended Command Syntax: Parameter Command

Prefix	Body (Subcommand)	Termination	Description
“AT”	+<name>=<value>|<compound values>	<CR><LF>	Store a <value> or <compound values>
“AT”	+<name>?	<CR><LF>	Read the current value

2.2 Response Syntax

Two types of response, the basic response and the extended response, are supported. The format used for these responses are shown in Table 9.

Table 9. General Response Syntax

Header	Information response	Trailer
<CR><LF>	Individual commands as specified	<CR><LF>

2.2.1 Basic Response

The basic response returns a result code in the information response as a result of basic and extended commands. Table 10 and Table 11 show the syntax of the basic response and the result codes respectively.

Table 10. Basic Response Syntax

Header	Result code	Trailer
<CR><LF>	<result code>	<CR><LF>

Table 11. Result Codes

Result code	Description
OK	Acknowledges execution of a command.
ERROR	Command not recognized, command line maximum length exceeded, parameter value invalid, or other problem with processing the command line.
BUSY	Other command is still running.

2.2.2 Extended Response Syntax

The extended response returns a subcommand information response in the information response as a result of extended commands. Table 12 shows the syntax of the extended response. The subcommand information response returns a value or compound values. The compound values use a comma as a delimiter. The response of the subcommand information response may be different except the common format.

Table 12. Extended Response Syntax

Header	Subcommand Information Response	Trailer	Description
<CR><LF>	+<name>:<value>|<compound values>	<CR><LF>	The body has the executed command name and the result value or compound values.

Some of the commands return what are referred to as an extended result code as extended response value. Table 13 shows the list of extended result codes a command can return.

Table 13. List of Extended Result Code

Extended Result Code	Description
OK	Success
BUSY	MAC is busy
ACK_RECEIVED	Received an ACK to a confirmed data message
NO_ACK	Did not receive an ACK to a confirmed data message
NO_RESPONSE	Did not receive a response to a MAC command sent
JOIN_ACCEPTED	Received Join Accept in response to Join Request
JOIN_FAILED	Did not receive Join Accept in response to Join Request
BEACON_NOT_FOUND	Could not receive a Class B beacon
SERVICE_UNKNOWN	Unknown MAC service
PARAMETER_INVALID	Invalid parameter is specified
FREQUENCY_INVALID	Invalid radio frequency is specified
DATARATE_INVALID	Invalid data rate is specified
NO_NETWORK_JOINED	Device is not activated
LENGTH_ERROR	Payload is too long
REGION_NOT_SUPPORTED	Specified region is not supported

Extended Result Code	Description
SKIPPED_APP_DATA	Application data transmission is skipped
DUTYCYCLE_RESTRICTED	Transmission is under duty cycle restriction
NO_CHANNEL_FOUND	There is no channel to send data at current data rate
NO_FREE_CHANNEL_FOUND	Transmission channels are busy
BUSY_BEACON_RESERVED_TIME	Transmission aborted due to beacon reserved time
BUSY_PING_SLOT_WINDOW_TIME	Transmission aborted due to ping slot window
BUSY_UPLINK_COLLISION	Transmission aborted due to beacon time
MC_GROUP_UNDEFINED	Multicast group undefined
MAC_ERROR	MAC error
RADIO_FAIL	Radio driver error
RADIO_PARAMETER_INVALID	Radio parameter configuration invalid
UNKNOWN_STATUS	Unknown error

3. AT Command Sets

3.1 Basic AT Command Sets

3.1.1 Basic AT Command List

Commands	Description
AT	Tests whether to accept AT commands.
ATE	Enables / disables Echo-back. Echo-back is enabled in the initial state. (*1)
ATV	Changes the response format of the result code and response information.

*1) Unsupported for RA

3.2 Basic AT Command Reference

3.2.1 AT

Command	Result Code	Description
AT	• OK • ERROR	• Tests if the AT command interface is valid.

3.2.2 ATE

Command	Result code	Description
ATE<value> <value> 0: Echo-back is disabled (default) 1: Echo-back is enabled	• OK • ERROR	• Enables or disables the echo-back. • Echo-back is enabled by default.

Note: Unsupported for RA

3.2.3 ATV

Command	Result code	Description
ATV<value> <value> 0: Delimiter before result code and response information is omitted (default) 1: Delimiter before result code and response information is added	• OK • ERROR	• Set the response format of the result code and response information.

Category	ATV0	ATV1
Information response	<text><CR><LF>	<CR><LF> <text><CR><LF>
Result code	<text><CR><LF>	<CR><LF> <text><CR><LF>

3.3 Extended AT Command Sets

3.3.1 Extended AT Command List

Commands	Description
+VER	Show the version of this sample application
+RESET	Reset LoRaMAC stack or MCU
+SAVE	Save parameters to the data flash
+LOAD	Load parameters from the data flash
+REGION	Set/get Region

Commands	Description
+DEVEUI	Set/get device EUI (DevEUI)
+NETID	Set/get network identifier (NetID)
+CLASS	Set/get device class
+DEVADDR	Set/get device address (DevAddr)
+APPEUI	Set/get application identifier (AppEUI)
+NWKSKEY	Set/get network session key (NwkSKey)
+APPSKEY	Set/get application session key (AppSKey)
+ACTMODE	Set/get activation mode
+MTYPE	Set/get message type (confirmed / unconfirmed) of data messages to be sent
+JOIN	Activate the device according to the activation mode
+SEND	Send data message of character string
+SENDHEX	Send data message of hexadecimal
+ADR	Enable/disable ADR mode
+RSSI	Enable/disable RSSI display
+RX1DELAY	Set the delay for Rx Window 1 in ms.
+LINKCHK	Send LinkCheckReq command
+FPORT	Set/get port number (FPort) of data messages to be sent
+DCYCLE	Enable/disable duty cycle control
+DR	Set/get default data rate in case ADR is disabled
+DEVTIME	Send DeviceTimeReq
+BCONACQ	Initiate beacon acquisition
+PNGSLPERIOD	Set/get PingSlot Periodicity
+PNGSLINFO	Send PingSlotInfoReq
+GENAPPKEY	Set/get a new root key (GenAppKey)
+CHDEFMASK	Set/get channels default mask
+DEVNONCE	Set/get device nonce
+APPNONCE	Set/get application nonce (AppNonce) in case of LoRaWAN V1.0.3/V1.0.2 specification or join server nonce (JoinNonce) in case of LoRaWAN V1.0.4 specification
+DOWNFCNT	Set/get downlink frame counter
+UPFCNT	Set/get uplink frame counter
+RCVD	(Notification) Received a data message
+BCONRCVD	(Notification) Successfully received a Class B beacon
+BCONNORCVD	(Notification) Failed to receive a Class B beacon at an expected time interval
+BCONLOST	(Notification) Lost synchronization with Class B beacons
+DEBUG	Set/get debug mode (Debug purpose only)

3.4 Extended AT Command Reference

3.4.1 AT+VER

Command	Result code	Description
+VER?	<version string> OK	Read firmware version

3.4.2 AT+RESET

Command	Result code	Description
+RESET=<mode> <mode> 0: Re-initialize LoRaMAC 1: Reset MCU 7: Initialize settings by formatting data flash and reset MCU	OK ERROR	• Require the re-initialize LoRaMAC (<mode>=0) or S/W reset (<mode>=1) • When <mode> is other than 0, 1 and 7, this command returns ERROR

3.4.3 AT+SAVE

Command	Result code	Description
+SAVE	OK	• Save parameters to the data flash. • The parameters to save are region, device class, activation mode, AppKey, DevEUI, AppEUI, NwkSKey, AppSKey, GenAppKey, DevAddr, NetID, message type, FPort, Rx Window 1 delay, duty cycle control, default data rate, ADR, RSSI settings, channel default mask, and ping slot periodicity. • Also the following parameters are saved. — When the activation mode is OTAA: DevNonce and AppNonce(JoinNonce) — When the activation mode is ABP: Frame counters (uplink and downlink), channels, channel mask, data rate for ADR on, NbTrans, TX power, max duty cycle, RX1 DROffset, RX2 frequency, Rx timing delay, maximum EIRP, dwell time (uplink and downlink), and ping slot data rate.

3.4.4 AT+LOAD

Command	Result code	Description
+LOAD +LOAD=<option> <option> 0: Load from the data flash (same as “+LOAD”) 1: Load default parameters	OK	Load parameters from the data flash when the option is not specified or set to 0. Set default values to parameters when the option is set to 1.

3.4.5 AT+REGION

Command	Result code	Description
+REGION=<region> <region> Operation region 0: EU868 1: US915 4: AU915 6: AS923-Group1 (*Note1) AS923 (*Note1) 7: KR920 8: IN865 22: AS923-Group2 (*Note2) 23: AS923-Group3 (*Note2) 24: AS923-Group4 (*Note2) 30: AS923-Japan (*Note3)	• OK • ERROR	• Set operation region to the device. • When the <region> is other than the value for supported region, this command returns ERROR.
+REGION?	<region> OK	Read the operation region set in the device.

Note1: AS923 feature is utilized when LoRaWAN V1.0.3 which encompasses V1.0.2 is enabled. Group AS923-1 feature is utilized when LoRaWAN V1.0.4 feature is enabled.

Note2: Group AS923-2 (<region>=22), AS923-3 (<region>=23), and AS923-4 (<region>=24) can be specified when LoRaWAN V1.0.4 feature is enabled (see section 4.1.2).

Note3: AS923 and Japan specific features are utilized when LoRaWAN V1.0.3 which encompasses V1.0.2 is enabled. Group AS923-1 and Japan specific features are utilized when LoRaWAN V1.0.4 feature is enabled.

3.4.6 AT+DEVEUI

Command	Result code	Description
+DEVEUI=<deveui> <deveui> Device EUI (DevEUI) in 8 bytes hexadecimal value (16 characters)	• OK • ERROR	• Set DevEUI to the device. • When the length of <deveui> less than 8 bytes, pad with 0x00 from MSB. • When the length of <deveui> exceeds 8 bytes, this command returns ERROR.
+DEVEUI?	<deveui> OK	Read the DevEUI set in the device.

3.4.7 AT+CLASS

Command	Result code	Description
+CLASS=<class> <class> Device class 0: Class A (default) 1: Class B 2: Class C	• OK • ERROR	• Set the device class to the device. • When <class> is other than 0, 1 or 2, this command returns ERROR.
+CLASS?	<class> OK	Read the device class of the device.

Note1: If Class B is used, the device class shall be changed from A to B via AT+CLASS after the beacon acquisition is succeeded and the ping slot periodicity is set to the intended value.

Note2: If Class C is used, the device class shall be changed from A to C via AT+CLASS after the device successfully joined the network.

3.4.8 AT+DEVADDR

Command	Result code	Description
+DEVADDR=<devaddr> <devaddr> Device address (DevAddr) in 4bytes hexadecimal value (8 characters)	• OK • ERROR	• Set DevAddr to the device. • When the length of `<devaddr>` is less than 4bytes, pad with 0x00 from MSB. • When the length of `<devaddr>` exceeds 4 bytes, this command returns ERROR. • When activation of OTAA is succeeded, DevAddr is updated to a value assigned by a server.
+DEVADDR?	<devaddr> OK	Read the `DevAddr` set in the device.

3.4.9 AT+NETID

Command	Result code	Description
+NETID=<netid> <netid> Network indentifier (NetID) in 3 bytes hexadecimal value (6 characters)	• OK • ERROR	• Set NetID to the device. • When the length of `<netid>` is less than 3 bytes, pad with 0x00 from MSB. • When the length of `<netid>` exceeds 4 bytes, this command returns ERROR. • When activation of OTAA is succeeded, NetID is updated to a value assigned by a server.
+NETID?	<netid> OK	Read the NetID set in the device.

3.4.10 AT+APPEUI

Command	Result code	Description
+APPEUI=<appeui> <appeui>: Application identifier (AppEUI) in 8 bytes hexadecimal value (16 characters).	• OK • ERROR	• Set AppEUI to the device. • When the length of `<appeui>` is less than 8 bytes, pad with 0x00 from MSB. • When the length of `<appeui>` exceeds 8 bytes, this command returns ERROR.
+APPEUI?	<appeui> OK	Read the `AppEUI` set in the device.

3.4.11 AT+NWKSKEY

Command	Result code	Description
+NWKSKEY=<nwkskey> <nwkskey>: Network session key (NwkSKey) in 16 bytes hexadecimal value (32 characters).	• OK • ERROR	• Set `NwkSKey` to the device. • When the length of `<nwkskey>` is less than 16 bytes, pad with 0x00 from MSB. • When the length of `<nwkskey>` exceeds 16 bytes, this command returns ERROR.
+NWKSKEY?	• ERROR • <nwkskey> OK	• By default, `NwkSKey` cannot be read and ERROR is returned. • If the macro `APP_AT_KEY_READ_ENABLED` is defined, `NwkSKey` can be read (see section 4.1.2).

3.4.12 AT+APPSKEY

Command	Result code	Description
+APPSKEY=<appskey> <appskey> Application session key (AppSKey) in 16 bytes hexadecimal value (32 characters).	• OK • ERROR	• Set `AppSKey` to the device. • When the length of `<appskey>` is less than 16 bytes, pad with 0x00 from MSB. • When the length of `<appskey>` exceeds 16 bytes, this command returns ERROR.
+APPSKEY?	• ERROR • <appskey> OK	• By default, `AppSKey` cannot be read and ERROR is returned. • If the macro `APP_AT_KEY_READ_ENABLED` is defined, `AppSKey` can be read (see section 4.1.2).

3.4.13 AT+APPKEY

Command	Result code	Description
+APPKEY=<appkey> <appkey> Application key (AppKey) in 16 bytes hexadecimal value (32 characters).	• OK • ERROR	• Set `AppKey` to the device. • When the length of `<appkey>` is less than 16 bytes, pad with 0x00 from MSB. • When the length of `<appkey>` exceeds 16 bytes, this command returns ERROR.
+APPKEY?	• ERROR • <appkey> OK	• By default, `AppKey` cannot be read and ERROR is returned. • If the macro `APP_AT_KEY_READ_ENABLED` is defined, `AppKey` can be read (see section 4.1.2).

3.4.14 AT+ACTMODE

Command	Result code	Description
+ACTMODE=<mode> <mode> Activation mode 0: ABP 1: OTAA (default)	• OK • ERROR	• Set activation mode to the device. • When `<mode>` is other than 0 and 1, this command returns ERROR
+ACTMODE?	<mode> OK	Read activation mode set in the device.

3.4.15 AT+MTYPE

Command	Result code	Description
+MTYPE=<mtype> <mtype> Message type of data message 0: Unconfirmed data (default) 1: Confirmed data	• OK • ERROR	Set the message type of data messages to be sent, unconfirmed or confirmed.
+MTYPE?	<mtype> OK	Read the message type.

3.4.16 AT+JOIN

Command	Result code	Description	
+JOIN	• OK • ERROR • +JOIN: JOIN_ACCEPTED • +JOIN: JOIN_FAILED	• Request to activate the device according to the activation mode setting (ABP or OTAA). • In case of ABP mode, the device is activated if the result code `OK` is received. • In case of OTAA, the device is activated if the result code `OK` and `+JOIN: JOIN_ACCEPTED` are received. • In other cases, the device failed to be activated.	
		Extended result codes	
		JOIN_ACCEPTED	Succeeded to receive Join-Accept message in case of OTAA mode.
		JOIN_FAILED	Failed to receive Join-Accept message in case OTAA mode.

Note1: AT+JOIN shall not be issued when the LoRaWAN stack operates in Class B.
Note2: In case of OTAA, AT+JOIN switches the device class to Class A. If Class B is used, the device class needs to be set to Class B after the completion of the beacon acquisition (see section 4.3.7). If Class C is used, the device class needs to be set to Class C after the completion of the AT+JOIN (see section 4.3.8).

3.4.17 AT+SEND

Command	Result code	Description	
+SEND=”<data>” <data> Data of character string	• OK • ERROR • +SEND: OK • +SEND: NO_NETWORK_ JOINED • +SEND: LENGTH_ERROR • +SEND: ACK_RECEIVED- • +SEND: NO_ACK • +SEND: SKIPPED_APP_ DATA	• Request to send a data message of character string. • Only visible characters, except space (0x20) and double quote (0x22) and comma (0x2c), are permitted for the character string. • Message type, unconfirmed data or confirmed data, can be specified by `AT+MTYPE` command. • Port number can be specified by `AT+FPORT` command.	
		Extended result codes	
		OK	Sent unconfirmed data message
		ACK_RECEIVED	Sent confirmed data message and received ACK
		NO_ACK	Sent confirmed data message and failed to receive ACK
		NO_NETWORK_JOINED	The device has not been activated
		LENGTH_ERROR	Too long data message length
		SKIPPED_APP_DATA	Sent MAC commands only. Need to retry sending data message.

3.4.18 AT+SENDHEX

<table>
<tr><th>Command</th><th>Result code</th><th colspan="2">Description</th></tr>
<tr><td rowspan="8">+SENDHEX=<data>

<data>
Data of hexadecimal value</td><td rowspan="8">• OK
• ERROR
• +SEND: OK
• +SEND: NO_NETWORK_JOINED
• +SEND: LENGTH_ERROR
• +SEND: ACK_RECEIVED
• +SEND: NO_ACK
• +SEND: SKIPPED_APP_DATA</td><td colspan="2">• Request to send a data message of hexadecimal value.
• Message type, unconfirmed data or confirmed data, can be specified by AT+MTYPE command.
• Port number can be specified by AT+FPORT command.</td></tr>
<tr><td colspan="2">Extended result codes</td></tr>
<tr><td>OK</td><td>Sent unconfirmed data message</td></tr>
<tr><td>ACK_RECEIVED</td><td>Sent confirmed data message and received ACK</td></tr>
<tr><td>NO_ACK</td><td>Sent confirmed data message and failed to receive ACK</td></tr>
<tr><td>NO_NETWORK_JOINED</td><td>The device has not been activated</td></tr>
<tr><td>LENGTH_ERROR</td><td>Too long data message length</td></tr>
<tr><td>SKIPPED_APP_DATA</td><td>Sent MAC commands only. Need to retry sending data message.</td></tr>
</table>

3.4.19 AT+ADR

Command	Result code	Description
+ADR=<mode> <mode> ADR mode 0: OFF 1: ON (default)	• OK • ERROR	• Set ADR mode to the device • When <mode> is other than 0 or 1, this command returns ERROR
+ADR?	<mode> OK	Read ADR mode set in the device

3.4.20 AT+RSSI

Command	Result code	Description
+RSSI=<mode> <mode> RSSI display mode 0: OFF (default) 1: ON	• OK • ERROR	• Set RSSI display mode to the device • When RSSI display mode is ON, RSSI and SNR values will be displayed on reception of messages. • When <mode> is other than 0 or 1, this command returns ERROR
+RSSI?	<mode> OK	Read RSSI display mode set in the device

3.4.21 AT+RX1DELAY

Command	Result code	Description
+RX1DELAY=<delay> <delay> Delay for Rx Window 1 in decimal [msec]. Range: 500 – 50000 Default: 1000	OK ERROR	• Set <delay> [msec] to the delay for Rx Window 1 and <delay> + 1000 [msec] to the delay for Rx Window 2. • Default value is 1000. • When <delay> is out of range, this command returns ERROR.
+RX1DELAY?	<duration> OK	Read the delay for Rx Window 1 in ms.

3.4.22 AT+LINKCHK

Command	Result code	Description	
+LINKCHK	• OK • ERROR • +LINKCHK: <Margin>,<GwCnt> • +LINKCHK: NO_NETWORK_JOINED • +LINKCHK: NO_RESPONSE <Margin> Link margin in dB <GwCnt> Number of gateways that received the LinkCheckReq command	• Request to send LinkCheckReq command. • If LinkCheckAns is received in response to LinkCheckReq command, <Margin> and <GwCnt> are notified.	
		Extended result codes	
		NO_NETWORK_JOINED	The device has not been activated
		NO_RESPONSE	Failed to receive LinkCheckAns command

3.4.23 AT+FPORT

Command	Result code	Description
+FPORT=<port> <port> Port number (FPort) in decimal Range: 1 - 224 Default: 1	OK ERROR	• Set the FPort of data 18essages to be sent. • When <port> is other than 1 to 224, this command returns ERROR.
+FPORT?	<port> OK	Read the FPort of data messages to be sent.

3.4.24 AT+DCYCLE

Command	Result code	Description
+DCYCLE=<mode> <mode> Duty cycle mode 0: OFF 1: ON (default)	OK ERROR	• Set duty cycle operation mode to the device • When duty cycle operation mode is ON, message send is controlled by duty cycle limitation. • When <mode> is other than 0 or 1, this command returns ERROR
+DCYCLE?	<mode> OK	Read the duty cycle operation mode.

3.4.25 AT+DR

Command	Result code	Description
+DR=<dr> <dr> Default data rate Range: 0 - 15	OK ERROR +DR: DATARATE_INVALID	• Set the default data rate of data messages to be sent when ADR is off. • When <dr> is out of regional specification, this command returns ERROR. • This data rate is just a request to the stack. The actual data rate used to send messages could be changed by the stack.
		Extended result codes
		DATARATE_INVALID: Data rate is out of regional specification
+DR?	<dr> OK	Read the default data rate of data messages to be sent when ADR is off.

3.4.26 AT+DEVTIME

Command	Result code	Description
+DEVTIME	• OK • ERROR • +DEVTIME: OK • +DEVTIME: NO_RESPONSE	• Requests to send DeviceTimeReq
		Extended result codes
		OK: Received DeviceTimeAns in response to DeviceTimeReq
		NO_RESPONSE: Failed to receive DeviceTimeAns command

Note 1: AT+DEVTIME shall not be issued when the LoRaWAN stack operates in Class B.

3.4.27 AT+BCONACQ

Command	Result code	Description
+BCONACQ	• OK • ERROR • +BCONACQ: OK • +BCONACQ: BEACON_NOT_FOUND	• Initiates Class B beacon acquisition • Result of beacon acquisition attempt is displayed up to 128 seconds after initiation of beacon acquisition. • If this command is not following a successful execution of AT+DEVTIME, the device can stay in the reception state up to 128 seconds
		Extended result codes
		OK: Successfully received a Class B beacon
		BEACON_NOT_FOUND: Could not receive a Class B beacon

Note 1: AT+BCONACQ shall not be issued when the LoRaWAN stack operates in Class B.

Note 2: If GPS time is received via AT+DEVTIME, AT+BCONACQ makes the LoRaWAN stack open a receive window to acquire a beacon frame around the calculated beacon frame reception timing. If not, AT+BCONACQ makes the LoRaWAN stack open a receive window to acquire a beacon frame up to 128 seconds.

3.4.28 AT+PNGSLPERIOD

Command	Result code	Description
+PNGSLPERIOD =<Periodicity> <Periodicity> Periodicity of Class B unicast ping slots. Range: 4 to 7 Default: 7	• OK • ERROR	• Sets periodicity of unicast ping slots • Returns ERROR if `<Periodicity>` is not in the valid range of 0 to 7
+PNGSLPERIOD?	<Periodicity> OK	Reads current periodicity configuration

Note 1: `AT+PNGSLPERIOD` shall not be issued when the LoRaWAN stack operates in Class B.

3.4.29 AT+PNGSLINFO

Command	Result code	Description	
+PNGSLINFO	• OK • ERROR • +PNGSLINFO: OK • +PNGSLINFO: NO_RESPONSE	• Requests to send the `PingSlotInfoReq` command. • Parameter "Periodicity" of `PingSlotInfoReq` can be set by `AT+PNGSLPERIOD`. • Returns ERROR if `<Periodicity>` is not in the valid range of 0 to 7	
		Extended result codes	
		OK	Received `PingSlotInfoAns` in response to `PingSlotInfoReq`
		NO_RESPONSE	Did not receive `PingSlotInfoAns` command

Note: `AT+PNGSLINFO` shall not be issued when the LoRaWAN stack operates in Class B.

3.4.30 AT+GENAPPKEY

Command	Result code	Description
+GENAPPKEY=<genappkey> <genappkey> GenAppKey in 16 bytes hexadecimal value (32 characters).	• OK • ERROR	• Set GenAppKey to the device. • When the length of `<genappkey>` is less than 16 bytes, pad with 0x00 from MSB. • When the length of `<genappkey>` exceeds 16 bytes, this command returns ERROR.
+GENAPPKEY?	• ERROR • <genappkey> OK	• By default, `GenAppKey` cannot be read and ERROR is returned. • If the macro '`APP_AT_KEY_READ_ENABLED`' is defined, `GenAppKey` can be read (see section 4.1.2).

3.4.31 AT+CHDEFMASK

Command	Result code	Description
+CHDEFMASK=<chdefmask#0>, <chdefmask#1>,...,<chdefmask#n> <chdefmask#0…#n> Channels default mask in 2byte hexadecimal value (4 characters) units, separated by commas. In case of US915 and AU915: the 'n' is 4.	• OK • ERROR	• Set channels default mask. • When the number of entries in each region is set to be exceeded, this command returns ERROR. • This command can be used in such case that the LoRaWAN gateway for US915 and AU915 does not support full (64+8) channels but limited (8+1) channels. • The information set by this command is set to the default value of the LoRaWAN stack when the region is changed. • This command is supported for US915 and AU915. For other regions, it is unsupported. Example 1: AT+CHDEFMASK=00FF,0000,0000,0000,0001 • Channel 0 to 7 (125 kHz BW) and 64 (500 kHz BW) are enabled. Example 2: AT+CHDEFMASK=0000,0000,0000,FF00,0080 • Channel 56 to 63 (125 kHz BW) and 71 (500 kHz BW) are enabled.
+CHDEFMASK?	< chdefmask > OK	Read the channels default mask.

3.4.32 AT+DEVNONCE

Command	Result code	Description
+DEVNONCE=<devnonce> <devnonce> Device nonce in 2 bytes hexadecimal value (4 characters).	OK ERROR	• Set DevNonce to the device. • When the length of <devnonce> is less than 4bytes, pad with 0x00 from MSB. • When the length of <devnonce> exceeds 4 bytes, this command returns ERROR. • When the activation mode is OTAA, DevNonce is changed when to send JoinRequest even if DevNonce is set by this command. • The LoRaWAN server based on the LoRaWAN V1.0.4 specification checks the value of DevNonce if it is increased compared to the previous one during the OTAA procedure (that is, JoinRequest and JoinAccept message exchange). If it is not increased, the procedure fails. It has a chance of the Inconsistency between the server and the end device especially during the test or debugging. (For example, the DevNonce value in the non-volatile memory is erased.) In such cases, this command can be used to set a value more than or equal to the one the server has.
+DEVNONCE?	<devnonce> OK	Read the DevNonce value.

3.4.33 AT+APPNONCE

Command	Result code	Description
+APPNONCE=<appnonce> <appnonce> Application nonce in 3 bytes hexadecimal value (6 characters).	OK ERROR	• Set AppNonce (or JoinNonce) to the device. • When the length of <appnonce> is less than 3 bytes, pad with 0x00 from MSB. • When the length of <appnonce> exceeds 3 bytes, this command returns ERROR. • When the activation mode is OTAA, AppNonce is changed to the value of JoinAccept when it is received and valid even if AppNonce is set by this command. • The LoRaWAN stack based on the LoRaWAN V1.0.4 specification checks the value of AppNonce if it is not the same to the previous one during the OTAA procedure (that is, JoinRequest and JoinAccept message exchange). If it is same, the procedure fails. • The value of AppNonce will be changed every time the LoRaWAN server sends JoinAccept. So, usually it is unnecessary to set the value by using this command. It might be useful for test or debug purpose.
+APPNONCE?	<appnonce> OK	Read the AppNonce value.

3.4.34 AT+DOWNFCNT

Command	Result code	Description
+DOWNFCNT=<downfnct> <downfnct> Downlink frame counter in 4 bytes hexadecimal value (8 characters).	OK ERROR	• Set the downlink frame counter to the device. • When the length of <downfnct> is less than 4 bytes, pad with 0x00 from MSB. • When the length of <downfnct> exceeds 4 bytes, this command returns ERROR. • When a downlink frame received from the LoRaWAN server is valid, the downlink frame counter in the LoRaWAN stack is changed to the value of the received downlink frame even if it is set by this command. • The LoRaWAN stack checks the value of the down frame counter if it is increased. If it is not increased, the frame will be discarded. • It has a chance of the inconsistency between the server and the end device especially during the test or debugging in ABP mode. (for example, the server is initialized and loses the counter) • In such cases, this command can be used to set a value more than or equal to the previous one.
+ DOWNFCNT?	<downfnct> OK	Read the downlink frame counter value.

3.4.35 AT+UPFCNT

Command	Result code	Description
+UPFCNT=<upfcntt> <upfcnt> Uplink frame counter in 4 bytes hexadecimal value (8 characters).	OK ERROR	• Set uplink frame counter to the device. • When the length of <upfcnt> is less than 4 bytes, pad with 0x00 from MSB. • When the length of <upfcnt> exceeds 4 bytes, this command returns ERROR. • When an uplink frame is sent to a server, the uplink frame counter is incremented even if it is set by this command. • The LoRaWAN server checks the value of the uplink frame counter if it is increased compared to the previous one. If it is not increased, the frame will be discarded. It has a chance of the inconsistency between the server and the end device especially during the test or debugging in ABP mode (for example, the uplink frame counter value in the non-volatile memory is erased) In such cases, this command can be used to set a value more than or equal to the one the server has.
+ UPFCNT?	<upfcnt> OK	Read the uplink frame counter value.

3.4.36 +RCVD

Command	Description	
+RCVD: <data>,<port>,<slinfo> <data> Hexadecimal data <port> Port number in decimal <slinfo> Reception information (window type, multicast/unicast). See indication code list on the right.	• Notify reception of a data message • This command is notified only when the data message is received without an error.	
	<slinfo> indication code	
	0	Unicast data received in the first receive window (RX1) following uplink
	1	Unicast data received in the second receive window (RX2) following uplink
	2	Unicast data received in continuously open receive window of Class C
	3	Multicast data received in the continuously open receive window of Class C
	4	Unicast data received in a ping slot of Class B
	5	Multicast data received in a multicast ping slot of Class B.

3.4.37 +RSSI

Command	Description
+RSSI: <rssi>,<snr> <rssi> RSSI value <snr> SNR value	• Notify RSSI and SNR values of received data message and beacon when RSSI display mode is ON. • See AT+RSSI (section 3.4.20) about RSSI display mode.

3.4.38 +BCONRCVD

Command	Description
+BCONRCVD:<Time>,<InfoDesc>, <Info[6]> <Time> Timestamp in millisecond (ms) since January 6, 1980 00:00:00 UTC. <InfoDesc> “Information descriptor” in the GwSpecific field of the received beacon frame. <Info[6]> 6-byte “Information” in the GwSpecific field of the received beacon frame.	• Class B beacon reception notification • Notifies a successful Class B beacon reception • Unsuccessful beacon reception at an expected time interval is notified with +BCONNORCVD (see section 3.4.39) • If GwSpecific field in beacon is invalid (CRC for the optional part is incorrect), <infoDesc> and <Info[6]> are not notified.

3.4.39 +BCONNORCVD

Command	Description
+BCONNORCVD	• Notification • Notifies that the device could not receive a beacon frame at an expected time interval

3.4.40 +BCONLOST

Command	Description
+BCONLOST	• Notification • Notified that the device lost synchronization with Class B beacons. This indicates MAC layer fails to receive a beacon for 120 minutes.

3.4.41 AT+DEBUG

Command	Result code	Description
+DEBUG=<debugmode> <debugmode> Debug mode in 4 bytes hexadecimal value (8 characters). The default value is 00000100.	OK ERROR	Note: This command can be used for debug purpose only. • Set debug mode. One or more of the following debug mode bit(s) can be specified to `<debugmode>`. — 00000100 (bit 8): Enables Pseudo MCU low power operation (*Note1). It needs to be set if MCU cannot return from the low power mode before receiving UART data. — 00000001 (bit 0): Enables debug log of Tx/Rx data as the sniffer mode format of Renesas LPWA Studio [4]. (*Note2) — 00000002 (bit 1): Enables debug log of radio Rx. (*Note3) — 00000004 (bit 2): Enables debug log of radio Tx. (*Note4) — 00000008 (bit 3): Enables debug log of radio CCA. (*Note5) for example, `AT+DEBUG=00000103` Specifies bit 0, bit 1, and bit 8 to debug mode • This command can be used when `DEBUG_LORAMAC` and `DEBUG_RADIO` macro are defined (see section 4.1.2). • Default debug mode can be set from the value of `DEBUG_LORAMAC_DEFAULT_MODE` macro if it is defined (see section 4.1.2).
+ DEBUG?	<debugmode> OK	Read the debug mode.

Note1. When this debug mode 'Pseudo MCU low power operation' is enabled, the application program goes into the infinite loop waiting for interrupts when the application program is idle and needs to make the MCU go to the low power mode. The application program exits the infinite loop by the interrupts from the radio, timer and UART Rx. This debug mode emulates the low power mode without actually making the MCU go to the mode so that the application program can receive the AT commands.

When this debug mode is disabled, the application program makes the MUC to the low power mode (that is,STOP) when it is idle. In this case, the application program needs to make the MCU return from the low power mode by using application specific interrupt when to receive the AT commands.

Note2. Example to display Tx/Rx frames using Renesas LPWA Studio (In case of OTAA):

1. Start LPWA Studio. Select COM port and click **CONNECT**.
2. Select **Setup** > **LoRaWAN** > Check **Enable** of **LoRaWAN Packet Decoder**.
3. Select **Setup** > **LoRaWAN** > Set AppKey data in **AppKey**.
4. Click **Receive** > **Start**
5. Select **File** > **Debug**
6. Select **Terminal**. Input following command to display Tx/Rx frames for debug: `AT+DEBUG` = 101
7. Select **Terminal**. Input AT commands for the sample program
8. Select **Receive**. Tx/Rx frames from LoRaWAN stack will be displayed.

Note: It is necessary to select `JoinRequest` and `JoinAccept` messages to calculate the session keys to display the fields of the frames.

Note3. Output format: “*Rx:(modem), (bandwidth), (datarate), (timeout), (freq)”
modem: 0:GFSK, 1:LoRa, bandwidth: 4:128 kHz, 5:250 kHz, 6:500 kHz, datarate: SF(LoRa), Bitrate (GFSK)
timeout: Rx timeout in symbols (LoRa), Rx timeout in bytes (GFSK), freq: Frequency [Hz]

Note4. Output format: “*Tx:(modem),(bandwidth),(datarate),(timeout),(freq),(power)”
modem: 0:GFSK, 1:LoRa, bandwidth: 4:128 kHz, 5:250 kHz, 6:500 kHz, datarate: SF(LoRa), Bitrate (GFSK)
timeout: Tx timeout [msec], freq: Frequency [Hz], power: Tx power [dBm]

Note5. Output format: “*CCA:(modem),(bandwidth),(ccaresult),(timeout),(freq),(power)”
modem: 0:GFSK, 1:LoRa, bandwidth: Bandwidth [Hz], ccaresult: 1:channel free (idle), 0:channel busy
timeout: Rx timeout in symbols (LoRa), Rx timeout in bytes (GFSK)

4. Example operations of sample application

This section describes the example operations of the LoRaWAN stack sample application.

The section 4.1 describes the preparation required for the end device. The section 4.2 describes the preparation required for the LoRaWAN network server. The section 4.3 describes the example operations using the AT commands.

4.1 Preparation for End Device

The sample application needs to be built and programmed to the hardware you use.

4.1.1 Hardware Setup

As for how to setup the hardware required for the sample application, please refer to [7] and [8].

4.1.2 Configuration of Sample Application

Table 14 shows the major macros available for the configuration of the sample application. These macros can be specified in the project build option as needed.

Table 14. Macros Available for the Configuration of the Sample Application

Macro	Description	Default
LORAWAN_VERSION_1_0_4	Supports LoRaWAN protocol version 1.0.4 and LoRaWAN Regional Parameters RP002-1.0.3. (LORAWAN_VERSION_1_0_3 cannot be specified simultaneously.)	Defined
LORAWAN_VERSION_1_0_3	Supports LoRaWAN protocol version 1.0.3 which encompasses version 1.0.2, and LoRaWAN 1.0.3 Regional Parameters Revision A. (It can be omitted, i.e. default version is 1.0.3.)	Undefined
LORAMAC_CLASSB_ENABLED	Enables Class B feature	Defined
REGION_AS923	Enables AS923 feature [LoRaWAN 1.0.4 only] Enables all groups of AS923 (AS923-1, AS923-2, AS923-3, AS923-4, and AS923-1 for Japan)	Defined
REGION_EU868	Enables EU868 feature	Defined
REGION_US915	Enables US915 feature	Defined
REGION_IN865	Enables IN865 feature	Undefined
REGION_AU915	Enables AU915 feature	Undefined
REGION_KR920	Enables KR920 feature	Undefined
RP_USE_RADIO_CFG_CHECK	Enable the regulatory function for each region in Radio Driver. Refer to [3].	Defined
APP_AT_KEY_READ_ENABLED	Enables to read keys such as `AppKey`, `AppSKey`, `NwkSKey` and `GenAppKey` by using corresponding AT commands	Undefined
DEBUG_LORAMAC, DEBUG_RADIO	Enables the debug mode. Both `DEBUG_LORAMAC` and `DEBUG_RADIO` need to be set when to use the debug mode. The debug mode is necessary if MCU cannot wake up from low power mode by an interrupt before receiving UART data.	Defined

<table>
<tr><th>Macro</th><th>Description</th><th>Default</th></tr>
<tr><td>DEBUG_LORAMAC_DEFAULT_MODE=0xXXXXXXXX</td><td>Specifies the default debug mode with the ORed value of the following if necessary. For more details, refer to section 3.4.41.
<table>
<tr><td>0x00000100</td><td>Enables Pseudo MCU low power operation.</td></tr>
<tr><td>0x00000001</td><td>Enables debug log of Tx/Rx data as the Sniffer mode format of Renesas LPWA Studio (see [4]).</td></tr>
<tr><td>0x00000002</td><td>Enables debug log of radio Rx.</td></tr>
<tr><td>0x00000004</td><td>Enables debug log of radio TX.</td></tr>
<tr><td>0x00000008</td><td>Enables debug log of radio CCA.</td></tr>
</table>
This macro can be specified when DEBUG_LORAMAC and DEBUG_RADIO are defined.</td><td>Defined.
Set to 0x00000100</td></tr>
</table>

4.1.3 Building of Sample Application

As for how to build the sample application, please refer to [7] and [8].

4.1.4 Programing of Object Files to Code Flash Memory

The object files of the sample application need to write to the code flash memory of MCU. As for the operations for the flash programing, please refer to [7] and [8].

4.2 Preparation for LoRaWAN Network Server

An end device related information such as the region, channel plan, device class, activation mode (OTAA/ABP), device EUI, and application key needs to be configured in the LoRaWAN network server you use. Refer to [5] or [6] for an example of the configuration.

4.3 Example Operations of End Device

This section describes the example operation of end device such as the initial settings, activation, data transmission/reception.

4.3.1 Initial Setting

Parameters need to be set based on the information preconfigured in the LoRaWAN network server.

The required parameters are different for each activation mode (OTAA and ABP). The section 4.3.1.1 and section 4.3.1.2 describe the initial setting for OTAA and ABP respectively.

It is recommended to use OTAA because it could be more secure. In case of OTAA, the session keys are generated on each network joining time. In case of ABP, the session keys are not usually updated throughout the lifetime of the end device.

4.3.1.1 Initial Setting in Case of OTAA (Over The Air Activation) Activation Mode

The following command sequence is an example to initialize the device setting for OTAA. Parameters for each command are just samples and need to be modified according to application program specifications. This example sets following parameters, region, device class, activation mode, device EUI (`DevEUI`), application identifier (`AppEUI`) and application key (`AppKey`). A Class B device needs to start as Class A and later switch to Class B. Each read command (for example, `AT+REGION?`) is optional and can be omitted. Optionally parameters can be saved to the data flash after the settings.

AT+REGION=6

OK

AT+REGION?

+REGION: 6:AS923-Group1

OK

AT+CLASS=0

OK

AT+CLASS?

```
+CLASS: 0:CLASS_A
OK
AT+ACTMODE=1
OK
AT+ACTMODE?
+ACTMODE: 1:OTAA
OK
AT+DEVEUI=90F
OK
AT+DEVEUI?
+DEVEUI: 000000000000090F
OK
AT+APPEUI=10E
OK
AT+APPEUI?
+APPEUI: 000000000000010E
OK
AT+APPKEY=F0E
OK
AT+SAVE
OK
```

4.3.1.2 Initial Setting in Case of ABP (Activation By Personalization) Activation Mode

The following command sequence is an example to initialize the device setting for ABP. Parameter values for each command are just samples and need to be modified according to application program specifications. This example sets following parameters, region, device class, activation mode, device EUI (DevEUI), device address (DevAddr), network session key (NwkSKey) and application session key (AppSKey). A Class B device needs to start as Class A and later switch to Class B. Each read command is optional (for example, `AT+REGION?`) and can be omitted. Optionally parameters can be saved to the data flash after the settings.

```
AT+REGION=6
OK
AT+REGION?
+REGION: 6:AS923-Group1
OK
AT+CLASS=0
OK
AT+CLASS?
+CLASS: 0:CLASS_A
OK
AT+ACTMODE=0
```

OK

AT+ACTMODE?

+ACTMODE: 0:ABP

OK

AT+DEVEUI=910

OK

AT+DEVEUI?

+DEVEUI: 0000000000000910

OK

AT+DEVADDR=01020304

OK

AT+DEVADDR?

+DEVADDR: 01020304

OK

AT+NWKSKEY=30F

OK

AT+APPSKEY=40E

OK

AT+SAVE

OK

4.3.2 Activation

The following command sequence is an example to activate the device according to the activation mode setting (OTAA or ABP).

In case of OTAA, the device is activated if the result code `OK` and `+JOIN: JOIN_ACCEPTED` are received.

In case of ABP mode, the device is activated if the result code `OK` is received.

In other cases, the device failed to be activated.

(In case of OTTA mode)

AT+JOIN

OK

+JOIN: JOIN_ACCEPTED

(In case of ABP mode)

AT+JOIN

OK

4.3.3 Send Unconfirmed Data of Character String

The following command sequence is an example to send an unconfirmed data message. This example sends an unconfirmed data message of “HELLO”, and receives data of 0xAA, 0x00, 0xBB, 0x11 and port number of 123 sent from a server. Note that the server must be configured to send a downlink data message before the `AT+SEND` command is executed in order to receive data.

AT+MTYPE=0

OK

AT+MTYPE?

+MTYPE: 0:UNCONFIRMED_DATA

OK

AT+SEND="HELLO"

OK

+SEND: OK

+RCVD: AA00BB11,123,0,0

4.3.4 Send Confirmed Data of Character String

The following command sequence is an example to send a confirmed data message. This example sends a confirmed data with payload ‘‘HELLO’’, and receives an ACK for the confirmed data message, data of 0xAA, 0x00, 0xBB, 0x11 and port number of 1, sent from a server.

AT+MTYPE=1

OK

AT+MTYPE?

+MTYPE: 1:CONFIRMED_DATA

OK

AT+SEND="HELLO"

OK

+SEND: ACK_RECEIVED

+RCVD: AA00BB11,1,0

4.3.5 Send Confirmed Data of Hexadecimal Value

The following command sequence is an example to send a data message of hexadecimal data sequence. This example sends a confirmed data message of 0xAA, 0x00, 0xBB and receives an ACK for the confirmed data sent from a server.

AT+MTYPE=1

OK

AT+MTYPE?

+MTYPE: 1:CONFIRMED_DATA

OK

AT+SENDHEX=AA00BB

OK

+SENDHEX: ACK_RECEIVED

4.3.6 Display RSSI on Receiving a Data Message

The following command sequence is an example to display RSSI. This example sends a confirmed data message of “HELLO”, and receives an ACK for the confirmed data message, data of 0xAA, 0x00, 0xBB, 0x11 and port number of 1, and RSSI and SNR on receiving it.

AT+RSSI=1

OK

AT+RSSI?

+RSSI: 1:RSSI_ENABLED

OK

AT+MTYPE=1

OK

AT+MTYPE?

+MTYPE: 1:CONFIRMED_DATA

OK

AT+SEND=”HELLO"

OK

+SEND: ACK_RECEIVED

+RCVD: AA00BB11,1,0

+RSSI: -59,10

4.3.7 Switching to Class B (Optional)

If the Class B feature is necessary, the device class needs to be changed from Class A to Class B. The following example command sequence shows the device class transition to Class B. In this example, the device switches to Class B after `DeviceTimeReq/DeviceTimeAns` command exchange and beacon acquisition.

AT+DEVTIME

OK

+DEVTIME: OK

AT+BCONACQ

OK

+BCONACQ: OK

+BCONRCVD: 1234567890000,00,000000000000

AT+PNGSLPERIOD=5

OK

AT+PNGSLINFO **;** Required when a period of ping slots is not set in server configuration.

OK

+PNGSLINFO: OK

AT+CLASS=1

OK

AT+CLASS?

+CLASS: 1:CLASS_B

OK

AT+LINKCHK ; Notifies LoRaWAN server that currently it is operating in Class B

; LinkCheckReq command which requests response is used for example.

OK

+LINKCHK: 23,1

4.3.8 Switching to Class C (optional)

If the Class C feature is necessary, the device class needs to be changed from Class A to Class C. The following example command sequence shows the device class transition to Class C. You need to switch the device class to Class C after AT+JOIN command.

AT+JOIN

OK

+JOIN: JOIN_ACCEPTED

AT+CLASS=2

OK

AT+CLASS?

+CLASS: 2:CLASS_C

5. Execution of the Certification Test Program for LoRaWAN

This section describes how to run the certification test program for LoRaWAN. The sample program can be used for the LoRaWAN certification test in a test house or the LoRaWAN pre-certification test using the LoRaWAN Certification Test Tool (LCTT).

The section 5.1 describes the preparation required for the end device. The section 5.2 describes the preparation required for the LoRaWAN network server. The section 5.3 describes the example operations using the AT commands.

5.1 Preparation for End Device

The sample application needs to be built and programmed to the hardware you use.

5.1.1 Hardware Setup

As for how to setup the hardware required for the certification test program, please refer to [8]. However, only the RA environment is supported.

5.1.2 Configuration of the Certification Test Program

Table 15 shows the macro available for the configuration of the certification test program. These macros can be specified in the project build option as needed. For other macro definitions, see section 4.1.2. Defining this macro enables the command to execute the certification test program in Table 16.

Table 15. Macros Available for the Configuration of the Certification Test Program

Macro	Description	Default
APP_COMPLIANCE	Enables execution of the certification test program	Undefined

Table 16. Commands Enabled in the Certification Test Program

Command	Result code	Description
+COMPLIANCE +COMPLIANCE=<option> <option> 0: The certification test program is disabled. 1: The certification test program is enabled.	OK BUSY	• Run the certification test program. • Options are saved to the data flash by the AT+SAVE command. • When option = 1 is set, the certification test program is executed after AT+RESET.

5.1.3 Building of Sample Application

As for how to build the sample application, please refer to [8].

5.2 Preparation for LoRaWAN Network Server

This certification test program has been tested using the LoRaWAN Certification Test Tool (LCTT). It supports the following conditions.

- In case of LoRaWAN v1.0.4, Class A and OTAA
  AS923, AU915, EU868, IN865, KR920, US915
- In case of LoRaWAN v1.0.3 which encompasses v1.0.2, Class A and OTAA
  AS923, EU868, IN865, KR920, US915

An end device related information such as the region, channel plan, device class, activation mode, device EUI, and application key needs to be configured in the LoRaWAN network server you use. Please refer to the respective manuals for configuration instructions. To obtain the LCTT, please refer to the following:

https://lora-alliance.org/lorawan-certification/

The following is an example of a connection in a test environment.

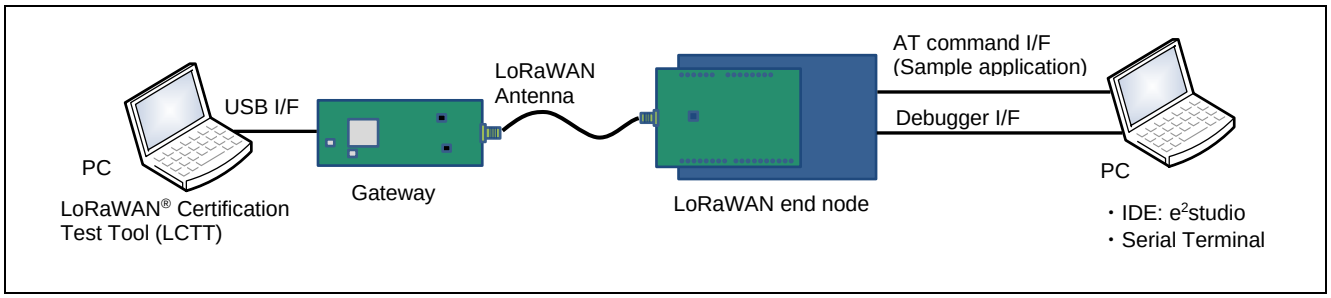

Figure 3. Test Environment Connection Example

5.3 Example Operation of End Device

This section describes the example operation of end device such as the initial setup and execution procedures.

5.3.1 Initial Setting

Parameters need to be set based on the information preconfigured in the LoRaWAN network server.

The required parameters are described in section 5.3.2.

5.3.2 Example Operations of the Certification Test Program

The following setting example shows the certification test program running on AS923, Class A and OTAA. In addition, `AT+DCYCLE=0` is set to reduce the execution time of the certification test program. Note that before running the application on the LoRaWAN end node, you need to run the LCTT.

When the certification test program is finished running, run `AT+RESET=7` to initialize the data flash.

AT+REGION=6

OK

AT+REGION?

+REGION: 6:AS923-Group1

OK

AT+CLASS=0

OK

AT+CLASS?

+CLASS: 0:CLASS_A

OK

AT+ACTMODE=1

OK

AT+ACTMODE?

+ACTMODE: 1:OTAA

OK

AT+DCYCLE=0

OK

AT+DCYCLE?

+DCYCLE: 0:DCYCLE_DISABLED

OK

AT+DEVEUI=90F

OK

AT+DEVEUI?

+DEVEUI: 000000000000090F

OK

AT+APPEUI=10E

OK

AT+APPEUI?

+APPEUI: 000000000000010E

OK

AT+APPKEY=F0E

OK

AT+COMPLIANCE=1

OK

AT+COMPLIANCE?

+COMPLIANCE: 1:COMPLIANCE_TEST_ENABLED

OK

AT+SAVE

OK

AT+RESET=1

OK

6. Limitation to Sample Application for RL78/G22

6.1 Command Line

In section 2.1, available number of characters for input of the command line is described.

When using RL78/G22, this sample application program can support up to 256 characters due to the internal RAM limitations.

The maximum number of characters can be configured by changing the value of `APP_AT_OCT_BUFF_SIZE` macro (default value is 256). It is specified in the project build option (see section 4.1.2).

6.2 AT Commands

When using RL78/G22, ABP, Class B functions, and multicast are not supported. So, there are some limitations to the following AT commands.

Table 17. limitations on AT commands

Commands	Section	Description
AT+SAVE	3.4.3	Following parameters are not saved because ABP is not supported. `NwkSKey`, `AppSKey`, `GenAppKey`, `DevAddr`, `NetID`, and `ping slot periodicity`.
AT+CLASS	3.4.7	1 (`Class B`) cannot be set because class B is not supported.
AT+DEVADDR	3.4.8	`DevAddr` can be read but cannot be set because ABP is not supported.
AT+NETID	3.4.9	`NetID` can be read but cannot not be set because ABP is not supported.
AT+NWKSKEY	3.4.11	Not supported because ABP is not supported.
AT+APPSKEY	3.4.12	Not supported because ABP is not supported.
AT+ACTMODE	3.4.14	0 (ABP) cannot be set because ABP is not supported.
AT+SENDHEX	3.4.18	Maximum size of <data> is limited to 121Byte (242 hexadecimal characters) (see section 6.1).
AT+BCONACQ	3.4.27	Not supported because class B is not supported.
AT+PNGSLPERIOD	3.4.28	Not supported because class B is not supported.
AT+PNGSLINFO	3.4.29	Not supported because class B is not supported.
AT+GENAPPKEY	3.4.30	Not supported because multicast is not supported.
+RCVD	3.4.36	Following `<slinfo>` indication codes are not notified because class B and multicast are not supported. 3 (multicast data received in class C window), 4 (unicast data received in a ping slot), and 5 (multicast data received in class B window)
+BCONRCVD	3.4.38	Not supported because class B is not supported.
+BCONNORCVD	3.4.39	Not supported because class B is not supported.
+BCONLOST	3.4.40	Not supported because class B is not supported.

7. Limitation to the Certification Test Program for RL78/G22

7.1 AT Commands

AT Command except “AT+COMPLIANCE” cannot be used when the certification test program is running.
When use all AT Command, stop the certification test program by executing AT+COMPLIANCE=0.

7.2 Log Output

Due to the limited ROM size, no logs can be output while the certification test program is running.

Revision History

Rev.	Date	Description	
		Page	Summary
01.00	Jan.31.19	-	First official release
01.10	Nov.29.19	-	Changed target device from 'RL78/G14 (R5F104JJ)' to 'RL78/G14 (R5F104ML)'
		2.2.2	Added table for 'List of Extended Result Code'.
		3.4.1.19, 4.7	Added SNR value when to display RSSI value
02.10	Jul.10.20	-	Supported Class B related commands and extended result codes
		1.3	Changed setting of local echo back to 'Yes'.
		2.1	Changed 127 characters to 512 characters for input of the command line
		3.2.2	Disabled echo back as default
		3.2.3	Omitted delimiter before result code and response information as default
		4.4	Added sample operation for Class B
03.00	Mar.26.21	-	Supported RL78/G23 (R7F100GLG) as a target device
03.10	Sep.30.21	-	Changed to support LoRaWAN protocol specified in the specification version 1.0.4.
		-	Supported RL78/G23 (R7F100GSN) as a target device
		3.4.2 3.4.32 - 35	Added AT commands: +RESET, +GENAPPKEY, +CHDEFMACK, +DEVNONCE, +APPNONCE, +DOWNFCNT, +UPFCNT
		3.4.3	Updated description of +SAVE
		3.4.5	Added operation region; AS923-2, AS923-3, AS923-4, and Japan.
		3.4.11 - 13 3.4.30	Changed that key cannot be read by default; +NWKSKEY, +APPSKEY, +APPKEY, +GENAPPKEY
		3.4.41	Added AT command for debug purpose: +DEBUG
		4.1	Added a section for the preparation for end device
		4.2	Added a section for the preparation for LoRaWAN network server
03.12	Jan.21.22	Table 1-2	Removed 'Real Time Clock (RTC)' for correction. Added I/O ports use in Application layer for correction.
		Table 1-3	Removed 'Real Time Clock (RTC)' for correction. Added I/O ports use in Application layer for correction.
		Table 1-4	Added I/O ports use in Application layer for correction.
		Table 4-1	Added a column for default settings. Added 'LoRaWAN Regional Parameters RP002-1.0.3' for LORAWAN_VERSION_1_0_4. Added 'LoRaWAN 1.0.3 Regional Parameters Revision A' for LORAWAN_VERSION_1_0_3. Added 'AS923-1 for Japan' for REGION_AS923.
04.00	Aug.29.22	-	Supported RA2E1 (R7FA2E1A9xxFM) as a target device
		3.4.5	Added operation regions; IN865, AU915, KR920.
		3.4.31	Added region AU915.
		4.3	Updated example operations according to the default setting of the ATV command.
		5	Added description of compliance test program for LoRaWAN

04.10	Nov.29.22	Table 4	Removed restrictions about supported regions. Updated restrictions about certification test feature
		Table 14	Removed restriction; RP_USE_RADIO_CFG_CHECK
04.20	Mar.31.23	-	Supported RA2L1 (R7FA2L1AB2DFP) as a target device
		3.4.7	Added note and described about switching to Class C.
		3.4.16	Added note and described about switching to device class.
		4.3.8	Added section and described the example of switching to Class C.
04.30	Jun.30.23	-	Supported RL78/G22 (R7F102GGE) as a target device
		6	Added section and described the limitation in case of RL78/G22.
		7	Added section and described the limitation to the Certification Test Program for RL78/G22
04.40	Dec.22.23	1.6	Replaced related document [3].
		Table 14	Updated description about RP_USE_RADIO_CFG_CHECK.
		4.3.3 4.3.4 4.3.6	Corrected typo.
		4.3.6	Modified description and command sequence.

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.

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

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/.

Trademarks

Arm® and Cortex® are registered trademarks of Arm Limited. Semtech, the Semtech logo, LoRa, LoRaWAN and LoRa Alliance are registered trademarks or service marks, or trademarks or service marks, of Semtech Corporation and/or its affiliates. Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.