

RA/RX/RL78 Family

Motor Control Development Support Tool Renesas Motor Workbench 3.2.0 Quick Start Guide

Introduction

Renesas Motor Workbench is a development support tool for Renesas MCUs for motor control (for target MCUs, see '1.3 Specifications') that performs debugging and tuning of motor control programs. It provides internal tools, Analyzer, Easy, and Tuner. This Quick Start Guide describes the procedure for Renesas Motor Workbench from installation to execution. For each function of Renesas Motor Workbench, refer to the following manual.

• Renesas Motor Workbench User's Manual (R21UZ0004)

Table of Contents

1.	Overview	2
1.1	About Renesas Motor Workbench	2
1.2	System Requirement	3
1.3	Specifications	5
1.4	Supportive Information (Reference)	6
2.	Installation and Execution	7
2.1	Download	9
2.2	Package Contents	9
2.3	Installation	10
2.4	Starting Software	11
2.5	Writing Execution Program	12
2.6	Loading Variable Information	18
2.7	Connecting Host PC and Evaluation Board	23
2.8	Description of Operation	29
2.9	Saving RMT file and terminating Renesas Motor Workbench	
3.	How to Generate Map File	
3.1	[RX] In CS+ (CC Compiler) Environment	34
3.2	[RX] In e ² studio Environment	
3.3	[RA] In e ² studio Environment	
4.	Communication Library	41
4.1	Communication Library for each MCU	41
4.2	Built-in Type Communication Library	47
4.3	How to Set User Program	
Rev	/ision History	54



1. Overview

1.1 About Renesas Motor Workbench

Renesas Motor Workbench is a tool that enables real-time reading/writing values of global variables of the program executed on the motor control evaluation board connected to a PC via USB.

Table 1-1 Main functions o	f Renesas	Motor Workb	ench
----------------------------	-----------	-------------	------

Function	Overview
Analyzer	 Starts when the "Analyzer" icon is clicked from Select Tool in Main Window. For global variables of the program executed on a board connected via USB, it can "load variable values", "write variable values", and "display variable values in waveforms in real time".
Easy GUI	 Starts when the "Easy" icon is clicked from Select Tool in Main Window. The GUI allows you to easily instruct drive and command values for programs executed on a board connected via USB.
Servo	 Starts when the "Servo" icon is clicked from Select Tool in Main Window. The GUI allows you to easily instruct servo for programs executed on a board connected via USB.
Tuner	 Starts when the Tuner icon is clicked from Select Tool in Main Window. Allows you to operate "measurement of peculiar parameter" and "auto-adjustment of control parameter ^{Note} of the motor for the program executed on a board connected via USB.

Note: The program executed on the board must have the auto-adjustment function.

Connection		File Inforn	nation		
СОМ	COM3 Clock	RMT File	RA6T1_MRSSK2_SPM_LESS_FOC_TUNEF	RV 4/5/2023 2:54:08 PM	
Status	Connect USB Serial Device	Map File	-	-	
Configuration		Select Too	I		
CPU	RA6T1	» Too and too		1967 A.	
Motor Type	Brushless DC Motor				
Control	Software for Tuner(Speed control)			10	
Inverter	RSSK for Motor	Eas	y Analyzer T	uner	
Project File Path D	Work\RA6T1_MRSSK2_SPM_LESS_FOC_TUNER_RV20)	V Q	Details	▼
Name			Date Modified	Size	
RAGT1_MRSSK2_SPM	LESS_FOC_TUNER_RV200.rmt		2023/04/05 14:54:08	279 KB	

Figure 1-1 Main Window of Renesas Motor Workbench

Renesas Motor Workbench assumes the following sample programs as the target.

- The Sample program included in this package
- The Sample programs provided by Renesas (downloadable from Renesas Web site)
- User programs



1.2 System Requirement

1.2.1 Target System

Renesas Motor Workbench can be used on the motor control board connected to a PC via USB. There are three types of boards depending on connection to the PC.

Type 1: Motor control inverter board that incorporates the communication function to the PC (isolated type)

(ex.) Renesas Solution Starter Kit - Evaluation System for BLDC Motor (Motor RSSK)

- Type 2: Motor control inverter board that connects to the PC with a communication board for tools ^{Note} (ex.) Renesas Flexible Motor Control Kit MCK-RA6T2 (including a communication board for tools)
- Type 3: Motor control inverter board that connects to the PC with a commercially available USB serial conversion module (isolated type). In this case, use the built-in type communication library (see 4. Communication Library.)
- Note: Communication board for tools: MC-COM (Renesas, RTK0EMXC90S00000BJ) or W2002 (Desk Top Laboratories Inc.)



Figure 1-2 System configuration when using Renesas Motor Workbench





Figure 1-2)

RENESAS

1.2.2 Host PC

Table 1-2 System requirement for Host PC

Item	Specification
PC	Windows PC with a USB terminal
	Available disk capacity: 150 MB and more
	RAM: 4GB and more
	USB ports: 1 ch (USB 2.0)
OS	Windows10, 11
.NET	Microsoft .NET Framework 4.6.1 or higher
Application	Adobe Reader (required when using PDF documents and Report function)
	Microsoft Excel (required when using CsvEdit function in Commander window)

1.3 Specifications

Table 1-3	Renesas	Motor	Workbench	specifications
-----------	---------	-------	-----------	----------------

Item		Specification
Target system	Supporte d MCU	RL78/G14, RL78/G1F, RL78/G24 RX13T, RX23T, RX24T, RX24U, RX26T, RX66T, RX72T, RX72M RA4T1, RA6T1, RA6T2, RA6T3, RA8T1
	Peripheral function	UART 1 ch, DTC (excluding RA MCUs), Port: TXD, RXD
Communicatio n function	Target system	Motor control inverter board with built-in type communication circuits (ex. Motor RSSK)
		 Motor control inverter board connected with a communication board for tools (ex. MCK-RA6T2) [Communication board for tools]] MC-COM (RTK0EMXC90S00000BJ, by Renesas) W2002 (by Desk Top Laboratories Inc. ^{Note)} Motor control inverter board connected with a commercially available USB serial conversion module (isolated type) (built-in type communication library is required.)
	PC	USB2.0
Internal function	Variable read/write	Maximum simultaneous selection: 255
	Waveform display	 Maximum simultaneous display: 12ch Maximum number of data 1024 (Motor RSSK, when using a built-in type communication library) 100000 (when using a communication board for tools 'MC-COM')

Note: Desk Top Laboratories Inc (http://www.desktoplab.co.jp/)



1.4 Supportive Information (Reference)

Renesas web pages provide colorful information on motor control as follows. The page "Renesas Motor Workbench" introduces how to use Renesas Motor Workbench through videos.

- Motor Control
 <u>https://www.renesas.com/en/key-technologies/motor-control</u>
- Renesas Motor Workbench
 <u>https://www.renesas.com/en/software-tool/renesas-motor-workbench</u>

RENESAS Products Applications Design Resource	s Support Sample & Buy About				
Design Resources / Development Tools / Renesas Motor Workbench					
Renesas Motor Workbench					
Solution Toolkit					
Motor Control Development Support Tool Renesas Motor Workbench 3.2.0 Qu Motor Control Development Support Tool Renesas Motor Workbench 3.2.0 Qu Motor Control Development Support Tool Renesas Motor Workbench 3.2.0 Use	ck Start Guide r's Manual				
Overview Downloads Documentation Design & Development	Survey Videos & Training Artificinal Datails				
Overview	Remease Motor Workbench provides powerful support for developers of mo data within programs.	for control applications, allowing operation of motor control programs from a PC and extraction of Example setup			
Description Features Release Information Target Devic Henesas Motor Workbarch is a development support tool for delogging analyzing. Easy GU Easy GU Interview	PC	Videos & Training			
Anaryzer Oynamic reading/writing of variables and waveform display while operating the	supporticol	videos & training			
motor.	Note: 30000 design	(1) Renesas Mohr Workbench (Setun)			
 Inset Automatic identification of motor parameters and control gains required for vector control, It's easy to the-tune and the results are instantly viable. 	Renesas Motor Workbench Functions		3 ⁴ 6	(1) Renesas Motor Workbench [Set-up] (2) Evaluation System for BLDC Motor [Set- up]	
	Servo implements an adjustment function for the motor's embedded p crigin return operation, and point-to-point control.)			(3) Motor Control Software [Download]	
	Functions needed for debugging are provided as APIs, allowing of Function Details			(4) Renesas Motor Workbench Analyzer [Basic]	
	Easy GUI Implements a GUI that allows more intuitive operation of the motor. Ability to set instruction values by manipulatine aliders.	connect the CPU card to the inverter board using the communication cable. (1) Renesas Motor Workbench [Set-up] Read More ~		(6) Renesas Motor Workbench Tuner [Basic]	Ŧ
You can learn various Motor Workbench	functions of Renesas n through videos.				

Figure 1-4 Renesas web page "Renesas Motor Workbench"



2. Installation and Execution







Figure 2-1 Flow from installation to execution of Renesas Motor Workbench



2.1 Download

Download Renesas Motor Workbench from the Renesas WEB. The downloaded file is compressed in .zip file, so unzip it in advance.

2.2 Package Contents

The folder construction of the downloaded package and the file list of each folder are shown below.

communication library ElfMapConverter Installer	 (1) Communication library (containing folders for each MCU) (2) Map file conversion tool (3) Installer and Authentication file (4) Execution file and environment file (containing folders for each evaluation board)
mot_rmt r20an0527ej0200-motor.pdf r20an0527jj0200-motor.pdf r20an0528ej0200-motor.pdf	
 reality2004/0000000000000000000000000000000000	(5) Documents (The name and type of included documents may differ.)

Figure 2-2 Folder construction of package (after unzipped)

Table 2-1 File list in each folder

No.	Folder	File	Description
(1)	Map file generation tool	ElfMapConverter.exe	Tool to generate a map file from elf file of RA MCU.
(2)	Communication library (containing folders for each MCU)	ICS_RX***.h ICS_RX***.obj ICS2_RX***.h ICS2_RX***.lib	File for RX MCU building
		ICS2_RL78***.h ICS2_RL78***.lib	File for RL78 MCU building
		ICS2_RA***.o ICS2_RA***_Built_in.o ICS2_RA***.h	File for RA MCU building (ICS2_RA***_Built_in.o is a built-in type communication library)
(3)	Installer	renesas_motor_workbench_****.msi	Renesas Motor Workbench installer
(4)	Execution file and RMT file	****.mot (RX, RL) ****.hex (RA)	Execution file to be written into Motor RSSK
	(containing folders for each evaluation board)	****.rmt	RMT file of Renesas Motor Workbench (file that saves environment settings)
(5)	Document	r21uz0004ej****-rmw-um.pdf r21uz0004jj****-rmw-um.pdf	Renesas Motor Workbench User's Manual
		r21qs0011ej****-rmw-qsg.pdf r21qs0011jj****-rmw-qsg.pdf	Quick Start Guide (this document)



2.3 Installation

To install Renesas Motor Workbench, execute installer "renesas_motor_workbench_***.msi" in the "Installer" folder.

After the installer starts, follow the instructions on the screen and install it.

Do renesas_mot (*** indi	ouble-click or_workbench_***.msi cates version No.)
Renesas Motor Workbench - X	
Welcome to the Renesas Motor Workbench Setup	
The installer will guide you through the steps required to install Renesas Motor Workbench on your computer.	🕼 Renesas Motor Workbench — 🗆 🗙
	Select Installation Folder
	The installer will install Renesas Motor Workbench to the following folder. To install in this folder, click "Next". To install to a different folder, enter it below or click "Browse".
WARNING: This computer program is protected by copyright law and international treaties. Unauthorized duplication or distribution of this program, or any portion of it, may result in severe civil or criminal penalties, and will be prosecuted to the maximum extent possible under the law.	Eolder: C#Program Files (x86)#RenesasElectronics#Renesas Motor Wor Browse
< Back Next > Cancel	Disk Cost
	Everyone Just me
	< <u>B</u> ack <u>N</u> ext> Cancel
Confirm Installation	<
The installer is ready to install Renesas Motor Workbench on your computer.	r ∰ Renesas Motor Workbench — □ X
Click "Next" to start the installation.	Installation Complete
	Renesas Motor Workbench has been successfully installed. Click "Close" to exit.
(Dook Massa Council	
<u>v</u> ∑duk <u>re</u> kti≯ Lancei	Please use Windows Update to check for any critical updates to the .NET Framework.
	< <u>B</u> ack Cancel

Figure 2-3 Installation steps



2.4 Starting Software

When Renesas Motor Workbench is installed, a shortcut icon will be shown to a desktop. Double-click the icon for "Renesas Motor Workbench" to start it.



Figure 2-4 Shortcut icon for starting

To start Renesas Motor Workbench from the Windows start menu, select "Renesas Electronics" > "Renesas Motor Workbench".



Figure 2-5 Starting from start menu

You can also start Renesas Motor Workbench by double-clicking a RMT file in the Windows Explorer. In this case, it starts with the selected RMT file already loaded.



Figure 2-6 Starting by double-clicking RMT file

If you cannot start Renesas Motor Workbench, installation might be failed. Uninstall and install it again following the procedure described in the previous section.



2.5 Writing Execution Program

Write an execution file generated by building the project of the motor control program or the execution file included in this package (.mot or .hex) into the target MCU. To write the file, use the tool such as IDE (CS+ or e^2 studio) or "Renesas Flash Programmer (RFP)".

Note: The methods of connecting a PC to a CPU board and of supplying power differ between when the execution program is being written and when Renesas Motor Workbench is being used. For details, refer to the manuals for each evaluation system.

2.5.1 When building and writing sample program with IDE

This section describes how to write the sample program provided by Renesas Web by using Integrated Development Environment (IDE), e² studio. (The RA MCU is used as an example here.)

For details on building, refer to the Application Note attached with the sample program.

(1) Importing project

Get the sample program from Renesas Web and import it into e² studio as the following steps.



Figure 2-7 Importing project (e² studio)



(2) Setting toolchain

In the properties of the project, set the tool configuration for your environment.

> ∰ Binaries > ∭ Includes > @ ra	Configure > Source > Properties Alt+Enter >	in Project Explore [Properti	r and select es]. C/C++Build > Set	$rac{1}{2}$ = 1 states the second states the se	
Properties for RA6T1_ES8_	SPM_LESS_FOC_E2S_V101	- 🗆 X	Properties for RA6T1_ESB_SP	M_LESS_FOC_E2S_V101	- 🗆 X
type filter text	Settings	(→ → → * 8	type filter text	Settings	(
Buildens VCr+ Build Build Variables Environment Logging Tool Chain Editor ClC++ General Project Natures Project Natures Project Natures Project Natures Project Natures Aun/Debug Settings Task Tags > Validation	Configuration: Debug [Active] Tool Settings Tool Solution Name Architecture Prefix Suffix Compiler Gut Tools for ARAM Embedd Architecture ARAM (Aarch32) Prefix Compiler Gut Compiler G	 ✓ Manage Configurations At the Binary Parses	Builders VC(++Build Build Whidels Environment Logging Tool Chain Editor YC(++ Greenal Project Alerences Rereas QE Run/Debug Settings Tasil Tags) Mildation	Configuration: Debug [Active]	Manage Configurations Manage Configurations Binary Parsets For Parsets F

Figure 2-8 Setting toolchain

(3) Building

Build the project. Confirm that there are no errors in the build-result.

	Open Project Close Project Open FSP Configuration		25_V1 ~ 🐇	
010	Build All Build Configurations	Ctrl+Alt+B	1_ESB_SPM	\ V_LESS_FOC_E2S_V101
	Build Project	Ctrl+B		Console ×
	Build Working Set Clean Build Automatically	>	-	CDT Build Console [RA6T1_ESB_SPM_LESS_FOC_E2S_V101] Extracting support files 15:58:12 **** Build of configuration Debug for project RA6T1_ESB_SPM_LESS_FOC_E2S_V101 **** make -r -j12 all
P	Generate Javadoc		1	arm-none-eabi-sizeformat=berkeley "RA6T1_ESB_SPM_LESS_FOC_E2S_V101.elf"
e²	C/C++ Index Update All Dependencies Change Device Change Toolchain Version	> Alt+D		text data bss dec hex filename 32944 140 3748 36832 8fe0 RAGT1_ESB_SPM_LESS_FOC_E2S_V101.elf 15:58:12 Build Finished. 0 errors, 0 warnings. (took 229ms)
1	C/C++ Project Settings Properties	Ctrl+Alt+P		

Figure 2-9 Building project



(4) Writing into evaluation board

Connect a PC to the evaluation board with USB cable (when the CPU board has a J-Link interface), write the program into the evaluation board.

Run Window Help	Debug Configurations	- D X
Renesas Debug Tools >	Create, manage, and run configurations	
i⇒ Instruction Stepping Mode		
[3_ Move to Line (C/C++)	C 2 0 1 × C 7 • N	ame: RA6T1_ES8_SPM_LESS_FOC_E2S_V101.elf
Resume at Line (C/C++)	type filter text	🗈 Main 🏇 Debugger 🐌 Startup 🤴 Source 🔲 Common
Nesume	C/C++ Application C/C++ Remote Application	Project
3 Suspend	EASE Script	RAGTI_ESB_SPM_LESS_FOC_E2S_V101 Browse
Terminate	GDB nardware beougging GDB Simulator Debugging (RH850)	C/C++ Application: Debug/RA6T1 ESB SPM LESS FOC E2S V101.elf
4 Disconnect		Variables Search Project Browse
🚏 Resume Without Signa	C: RAGT1_ESB_SPM_LESS_FOC_E2S_V101.elf	Build (if required) before launching
Step Into		Ruild Configuration: Use Active v
3 Step Over configurations		Crabbeauto build O Disable auto build O Lisable auto build O Lisable auto build
Step Return		Contractive versings
Run to Line		
Use Step Filters Shift+F5		
Pup Ctd (E11		Confirm that the target project to
Debug Ett		debug is selected.
Debug FTT		
Run History >		
Run As	Filter matched 9 of 11 items	Regert Apply
Run Configurations		
Debug History >		<u>D</u> ebug Close
Is Debug As >		
Debug Configurations	Debug configurations > [Debugg	er] tab
Breakpoint Typer	Debug Configurations	— 🗆 X
breakpoint types	Create, manage, and run configurations	
	📑 🖻 🕭 🖹 🗶 🖻 🍸 🔹	Name: RAGT1_ESB_SPM_LESS_FOC_E2S_V101.elf
	type filter text	📄 Main 🕷 Debugger 🐌 Startup 🦆 Source 🔟 Common
	C C/C++ Application	Debug hardware: J-Link ARM V Target Device: R7FA6T1AD
	EASE Script GDR Hardware Debugging	
	C GDB Simulator Debugging (RH850)	GDB Connection Settings
	✓ CI Renesas GDB Hardware Debugging	Autostart local GD8 server Host name or IP address: localhost
	C RAGT1 ESB SPM LESS FOC E2S V101.elf	
	Renesas Simulator Debugging (RX, RL78)	lect the debugger. Confirm the target device.
	C Renesas Simulator Debugging (RX, RL78)	ect the debugger. Confirm the target device.
	Renesas Simulator Debugging (RX, RL78)	dect the debugger. Connection Confirm the target device.
	Renesses Simulator Debugging (RX, RL78) Se	older the debugger. OB pot m Confirm the target device. convection Confirm the target device. Convection convection GOB Browse. Variable.
	Revess Simulator Debugging (RX, R178) Se	G08 G08 port in Confirm the target device. G08 G08 G08 command: amm none eabi-gdb Browne. Variables.
	Remess Simulator Debugging (RX, RL78) Se	G08 G08 pot m Confirm the target device. G08 G08 command: am none-eabl-gdb Browse. G98 pot m Browse. Watables
	Reness Simulator Debugging (RX, RL7a) Se	dect the debugger. Confirm the target device. Go8 Go8 Go8 Go8 Gommand: arm none-eabl-gdb Step Mode
	Remess Simulator Debugging (RX, RL70 Se	Additional GDB Ser Click [Debug] to execute
	Remess Simulator Debugging (RX, RL70 Se	Additional GDB Ser Click [Debug] to execute program writing
	Remess Simulator Debugging (RX, RL70 Se	Additional G08 Ser Click [Debug] to execute progr.am writing
	Remeas Simulator Debugging (RX, RL7a) Se Filter matched 9 of 11 items	Additional G00 See Click [Debug] to execute progr.am writing
Confirm on the console	Preness Simulator Debugging (RX, RL70 Se Filter matched 9 of 11 items	Additional GDB See Click [Debug] to execute program writing
Confirm on the console screen that [Download	Perress Simulator Debugging (RX, RL70 Se Filter matched 9 of 11 items	Additional GDB Ser Click [Debug] to execute progr.am writing Reget April
Confirm on the console screen that [Download is completed] .	Perress Simulator Debugging (RX, RL70 Se Filter matched 9 of 11 items	lect the debugger. Confirm the target device. Connection Confirm the target device. GOB command:
Confirm on the console screen that [Download is completed] .	Preness Simulator Debugging (RX, PL70) Se Filter matched 9 of 11 items	Additional GDB Ser Click [Debug] to execute progr.am writing Petror Petror Petror Cone
Confirm on the console screen that [Download is completed] .	Perress Simulator Debugging (RX R170 Se Filter matched 9 of 11 items	lect the debugger. Confirm the target device.
Confirm on the console screen that [Download is completed] .	Perress Simulator Debugging (RX, RL70) Filter matched 9 of 11 Rems	lect the debugger. Confirm the target device. GOB command. Imm more eable gale GOB command. Imm more eable gale GOB Command. Imm more eable gale Additional GOB Ser Click [Debug] to execute progr.am writing Repert Apply Box Click [Debug] to execute Progr. Box C
Confirm on the console screen that [Download is completed] .	Premess Simulator Debugging (RX R170 Filter matched 9 of 11 items Thermaty are Debugging] [pid: 24]	Additional GOB Ser Click [Debug] to execute progr. am writing Rest Apply Rest Apply Rest Click [Debug] to execute progr. am writing Rest Apply Rest Click [Debug] to execute Rest Apply Rest Click [Debug] to execute Progr. am writing Rest Apply Rest Click [Debug] to execute Progr. am writing Rest Apply Rest Progr. am View Pro
Confirm on the console screen that [Download is completed] . lole × IIII Registers III Debugger Console II M SB_SPM_LESS_POC_ESS_V101.elf [Reness GDB Hardw ed target_fonnection 2955	Perress Simulator Debugging (RX, RL70) Filter matched 9 of 11 items ?	elect the debugger. Confirm the target device. Convector Convector Convector Confirm the target device. GOB Command: Immone eabingth Step Mode Click [Debug] to execute progr. am writing Repert Apply Debug Cose
Confirm on the console screen that [Download is completed] . ble X III Reysters Debugger Console M B SPM_LESS_POC_ESS_V101.elf [Renesas GDB Hardw d target connection 1955 connection status - 0K	Perreas Simulator Debugging (RX R170 Filter matched 9 of 11 items rane Debugging] [pid: 24]	elect the debugger, Consection Confirm the target device.
Confirm on the console screen that [Download is completed] . Je X IIII Reysters Debugger Console I M SSM_LESS POC_E2S_V101.eff [Renesas GDB Hardw d target donnection 955 connection status - OK connection status - OK g download	Perress Simulator Debugging (RX R170 Filter matched 9 of 11 Rems To bebugging] [pid: 24]	elect the debugger. Construction Confirm the target device.
Confirm on the console screen that [Download is completed] . le X WH Registers Debugger Console M M 8 SPM_LESS FOC ESS VIOLeff [Renesas GDB Hardw d target connection 955 connection status - OK connection status - OK g download g download Function Select, writing to address @xx	Perress Simulator Debugging (RX, R1, 70 Filter matched 9 of 11 items emory are Debugging] [pid: 24] Peress Simulator Debugging] [pid: 24]	Additional GOB See Click [Debug] to execute progr.am writing Beever
Confirm on the console screen that [Download is completed] . IN IN Registers Debugger Console M S SPM LESS FOC ESS VIOLelf (Renesas GDB Hardw d target connection S55 connection status - OK connection status - OK g download Function Select, writing to address 0x08080488 with d download	Primer matched 9 of 11 items Filter matched 9 of 11 items Transport (7) Itemory Itemory	Alect the debugger. Constraint the target device. Connection Connection Confirm the target device. Connection Connection Confirm the target device. Connection Confirm the target device. Withdeter Association Confirm the target device. Mathematical Confirm the target device. M
Confirm on the console screen that [Download is completed] . ole X IM Resters Debugger Console I M B SPM LESS FOC ESS VIOLeff [Renesas GDB Hardw ed target connection 2955 connection status - OK connection status - OK connection status - OK g download Function Select, writing to address 0x6 ox, writing to address 0x60000408 with ed download re breakpoint set at address 0x1130	Perress Simulator Debugging (RX, RL 70) Filter matched 9 of 11 items Thermory are Debugging] [pid: 24]	lect the debugger. Convector Confirm the target device.

Figure 2-10 Writing program into evaluation board



(5) Disconnecting evaluation board

After writing the program, select "Run" > "Disconnect" to disconnect the evaluation board and remove the USB cable.

Instruction Stepping Mode Nove to Line (Select Resume at Line (Select Select Sesume Vithout Signal Step Into Step Over Console × Debugger Consol cterminated > RAGT1_ESB SPM_LESS Ff GOB: 52955 Target connection status - 0K Target connection status - 0K	[Run]- nnect]. F8 Ctrl+F2 F5 F6				
Move to Line (Select Select	[Run]- nnect]. F8 Ctrl+F2 F5 F6				
Resume at Line (> [Disco Resume Suspend Terminate Console	F8 Ctrl+F2 F5 F6				
Resume Suspend Terminate Suspend Terminate Console X Conso	F8 Ctrl+F2 F5 F6				
Suspend Terming/e Terming/e Terming/e Sep Into Sep Over Console X	Ctrl+F2 F5 F6				
	Ctrl+F2 F5 F6				
	F5 F6				
Console X @ Debugger Consol Console X @ Debugger Consol terminated> RAGT1_ESB_SPM_LESS_FG SDB: 52955 Target connection status - OK Target connection status - OK	F5 F6				
 Step Into Step Over Console X Pobugger Consol terminated> RAGT1_ESB_SPM_LESS_FG S2955 Target connection status - OK Target connection status - OK 	F5 F6				
Step Over Console X Ge Debugger Consol tterminated>RAGT1_ESB_SPM_LESS_F4 S08: 52955 Target connection status - 0K Target connection status - 0K	F6				
Console X Robugger Consol <terminated> RAGT1_ES8_SPM_LESS_F4 080: 52955 Target connection status - 0K Target connection status - 0K</terminated>					
<pre><terminated> RA6T1_ESB_SPM_LESS_F(3DB: 52955 Farget connection status - OK farget connection status - OK</terminated></pre>	le	Confirm on console		■ 🗶 💥 🗟 🛃 🛃 🚰 🚰 📑 🗖 🗸	📑 🗕 🗖
GDB: 52955 Target connection status - OK Target connection status - OK	OC_E2S_V101.elf [Renesas GD	screen that the target	DB server (Host) (Terminated Apr 14, 2023, 5:23:45 PM) [pid: 24]		
Target connection status - OK	(device is disconnected .			^
0	č	L			
Starting download	ing to address avagage	100 with data ffffffffffff	45555		
SECMPUxxx, writing to address	s 0x00000408 with data	fcffffffffffffffffffffffffffff	ffffffff		
Finished download					
Hardware breakpoint set at ad Disconnected from the Target	Debugger				
Sisconnected from the furget	Debugger :				~

Figure 2-11 Disconnecting evaluation board

If the project has been rebuilt, load the Map file which was generated at the same time as the latest execution file into Renesas Motor Workbench. For how to generate a Map file, see chapter 3, and for how to load the file, see chapter 2.6.2.

Note: If executing the program (driving a motor) using Renesas Motor Workbench after writing the program, use a communication dedicated circuit (isolated type) to connect to the target board.



2.5.2 When writing with Renesas Flash Programmer (RFP)

This section describes how to write the execution file included in this package into the target board using Renesas Flash Programmer (RFP). (You can download RFP from Renesas Web.) Configure settings for communication to the CPU board according to the system to be used.

Load the RMT file that is included as well. (For how to load RMT files, see chapter 2.6.1.)

- (1) Launch the RFP tool and create a new project.
- (2) Specify the communication method and connect a PC and the target board.

2	Renesas Flash Programmer V3.11.01	- 🗆 X
File	Target Device Help	
	New Project	Unique Code
	Open Project Save Project	Name the project.
	Save Image File	K Create New Project
	File Checksum	Project Information
	Set File Password	Microcontroller: RA ~
	1 RA6T1_MRSSK2_SPM_LESS_FOC_TU	Project Name: RA6T1_MRSSK2_SPM_LESS_FOC_TUNER_RV2
	2 RA6T1_MRSSK2_SPM_ENCD_FOC_TI 3 RA6T1_ESB_SPM_LESS_FOC_E2S_V10	Project <u>F</u> older: D¥Work <u>B</u> rowse
	Exit	Communication
-		<u>T</u> ool: J-Link
	Sta	Tool Details Num: AutoSelect
Renes	as Flash Programmer V3.11.01 [5 Jan 2b	Zoj
		Select the connection to the target. After connecting the target board and a PC with the communication method you specified, click [Connect].

Figure 2-12 Writing execution program with Renesas Flash Programmer (1/2)



- (3) Specify the program to write and start writing.
- (4) After writing is normally completed, terminate RFP and remove the target board from the PC.

Renesas Flash Programmer V3.11.01	- D X
File Target Device Help	
Operation Operation Settings Block Settings Connect	Settings Unique Code
Project Information Current Project: RA6T1_MRSSK2_SPM_LESS_FOC Microcontroller: RA Program File CProjectDir2#RA6T1_MRSSK2_SPM_LESS_FOC_TUNE	C_TUNER_RV200rpj Specify the program to write (.mot or .hex).
	CRC-32 : EC541B9F
Flash Operation	
Erase >> Program >> Verify	
<u>S</u> tart	ок
Connecting to the target device Connecting to the target device Signature: Device: RA Device Code: 00270447 Setting the target device Disconnecting the tool Operation completed.	After preparation, click [Start].
	Clear status and message
During writing	Writing is completed (normal termination)
Z Renesas Flish Programmer V3.11.01 — 🗌 🗙	🖉 Renesas Risch Programmer V3.11.01 — 🗆 🗙
Elle Target Device Help Operation Operation Settings Block Settings Connect Settings Unique Code	Bile Target Device Help Operation Operation Stattings Block Settings Connect Sattings Unique Code
Project Information Current Project R46TL/MESK2.SPM_LESS_FOC_TUNER_RV200rp) Microcontroller R4	Project Enternation Connect Project: RAST1_MPSSK2_SPMLESS_FOO_TUNER_RV233(p) Manacentralian: RA
Prozeniii File (Prozentii File (Prozentii Prozentii MRSSK2, SPM_LESS_FOC_TUNER_R/2001exx) (CRC+00 - RC441188 Prozens Report Flash Operat	Program File (ChrowstellevMANTLMPESIt2.SPMLESS.FOC,TUNER_RV20Ihme ORD-12: ECK1ESF Filed: Operation
Erase >> F Purnie_	Ense >> Program >> Verify Start OK
Ensing the selected blocks (Code Fach 1) 6x00007FFF size :512 K (Code Fach 1) 6x000000 - 6x0017FFF size :512 K (Code Fach 1) 6x0010007 - 6x00101FFF size :51 K (Code Fach 1) 6x0010007FFF size :56 K (Code Fach 1) 6x001007FFF size :56 K (Code Fach 1) 6x010041FF size :16	EDWINE Areal but INFATUL - but INFATULE - super 15 A Verifying data EDWINE Fisch 10:00000000 - 0:000000FFF - size - NK K EDWINE Fisch 20:000000FFF - size - NK EDWINE - but INFATULE -
Verifying data (Code Fach 1) 8x0000000 - 8x00000FFF size = 56 K Citer status and message	Cherrotine completed.

Figure 2-13 Writing execution program with Renesas Flash Programmer (2/2)

Note: If executing the program (driving a motor) using Renesas Motor Workbench after writing the program, use a communication dedicated circuit (isolated type) to connect to the target board.



2.6 Loading Variable Information

Renesas Motor Workbench loads the Map file which is outputted in generation of the execution program and gains access to the global variables. If the program has been changed and rebuilt, it is necessary to load the Map file again.

Renesas Motor Workbench can save and load the variable information and environment information including settings for tools as an RMT file (An RMW project file).



Figure 2-14 Map file and RMT file (An RMW project file)

2.6.1 Loading RMT File

When an RMT file (storing both variable information and tool setting information) is provided in the sample program included in this package or in the sample program provided by Renesas, load the RMT file by following one of the operations below.

- (a) Use Select Navigation function (displayed from Help menu) and load the RMT file.
- (b) Select "Open RMT File" of the Main Window's File menu and load the RMT file.
- (c) Click the "..." button next to the RMT File field in Main Window's File Information and load the RMT file.
- (d) Specify the project folder in Main Window's Project File Path and double-click the RMT file displayed on the File List to load it.
- (e) Drug & drop an RMT file on a PC to Main Window's File List and doble-click to load it. (In this case, the RMT file is copied to the project folder on the PC.)
- Note that if the sample program has been changed and built, it is necessary to load the Map file generated in building into Renesas Motor Workbench again.



§ Safert Nergation X	
Welcome to the Select Navigation	
We will support the settings required to use	Sidestinguin X
Click "Next".	
will not appear from the next	Select RMT File
time.	Select the RMT file you want to use and click "Next". The RMT file is an environment file that saves various settings used in RMW.
	Selecting RMT: RA6T1_ESB_SPM_LESS_FOC_E2S_V101.rmt
Do not show this features in the future	
Next > Finish	Select the RMT file to load.
>Rm Select >End Assistance	Do not show this features in the future
	→
	< Back Next > Finish
	>Start >Confirmation >End Assistance
dect Navigation X	
Map File Select Confirmation	
Do you want to select a MAP file? RMT file is undated when man file is selected	
with the is updated when they he is service.	© Methoda Sector
Yes No	Select COM Port established, you can select a
If you want to load a Map file, click	Select the COM port that communicates with the ICS board and click "Next".
Do not show this reatures in the ruture	Selecting COM: COM3 Clock
< Back Next > Finish	Click [Finish] to complete
>Select Rmt >Confirmation >End Assistance	Do not show this features in the future setting.
	< Back Next > Finish
	Select MAP >Confirmation >End Assistance
Nergeton X	
Select Map File	Display
	commator screen.
Select the MAP fue and click INext .	€ Northagen x
Selecting MAP: RA6T1_ESB_SPM_LESS_FOC_E2S_V101_conv.map	Completion to use RMW Tool
	You have completed the necessary settings to use Renesss Motor Workbench (RMW), Click "Individ and setect the feature you want to use.
	Click [Finish] to complete
Do not show this features in the future	setting.
	Do not show this features in the future
< Back Next > Finish Select Science Comparison Select Science	Finish
Privit Select >End Assistance	>End Assistance

Figure 2-15 ① Loading RMT file using Select Navigation function





Figure 2-16 ② Loading RMT file from "Open RMT File" of File menu







Figure 2-18 (4)(5) Loading RMT file from Main Window's File List



2.6.2 Loading Map File

When using a user program (when an RMT file doesn't exist), you need to load a Map file into Renesas Motor Workbench first.

You must use the Map file that was created at the same time as the execution file in building. If the program has been changed and rebuilt, it is necessary to load the Map file again as well. For generating a Map file, refer to "3. How to Generate Map File".

Steps for loading Map file:

- Load a Map file by one of the operations below. (For how to operate, refer to the figures in 2.6.1 Loading RMT File.)
 - (a) Use Select Navigation function (displayed from Help menu) to load the Map file.
 - (b) Select "Load Variable Data" from the Main Window's File menu and specify the Map file to load.
 - (c) Click the "..." button next to the Map File filed in Main Window's File Information to load it.

For RX:

Load a file with a *.map extension.

For RA:

In the file selection dialog, select "All Files (*.*)" for file type, and load a file with a *.rmap extension. If "MAP Files (*.map)" is selected for file type, a file with *.rmap extension is not displayed.

However, you can change the extension from *.rmap to *.map by the method described in the Tips of "3.3 [RA] In e2 studio Environment" or by modifying the extension directly. In this case, load the file with *.map extension that you have modified without changing the file type in the file selection dialog.

🛞 Load Variable Data			×
\leftarrow \rightarrow \checkmark \uparrow \blacksquare \ll app \Rightarrow rmw	~	ບ ເພນ	検索
整理 ▼ 新しいフォルダー			₿≕ ▾ 💷 😮
名前	更新日時	種類	サイズ
RX72T_ESB_SPM_LESS_FOC_CSP_V110.map	2022/09/13 10:30	MAP ファイル	117 KB
			表示するファイルの種類 選択します
ファイル名(N):		 MAP Files (* MAP Files (* All Files (*.*) 	*.map) ~ .map)

Figure 2-19 Map file selection dialog box

(2) After loading the Map file, "Use Setting Form" screen is displayed. If "Change in Data Type" or "Change to Array" has been set in the Option Dialog (displayed by selecting "Option Dialog" from Option menu) in loading, these settings are reflected to the list.



– 🗆 X
- ~
Set
DataType 🔺
LOAT
LOAT
LOAT
LOAT
LOAT 👻
LOAT

Figure 2-20 User Setting Form screen

You can also change the Data Type settings on the User Setting Form screen.

🛞 User Settir	ng Form	- 0	×
		Set	
Address	Name	DataType	
1ffe0198	g_f4_id_ref_monitor	FLOAT	· _
1ffe01a8	g_f4_iv_ad_monitor	UINT8	
1ffe01ac	g_f4_iw_ad_monitor	INT8	
1ffe01f0	g_f4_vdc_ad_monitor	UINT16	
1ffe01ec	g_f4_vd_ref_monitor	INT16	-
		UINT32	
		INT32	
		FLOAT	

Figure 2-21 User Setting Form screen (change Data Type settings)

- (3) When you press "SET" button, the variable information is loaded into Renesas Motor Workbench.
- Note that after loading variable information by pressing "SET" button, you cannot change the Data Type using the function of Renesas Motor Workbench. To make a change, load the variable information with "Load Variable Data" function again.



2.7 Connecting Host PC and Evaluation Board

2.7.1 Connecting via USB

For system requirements when Renesas Motor Workbench is used, refer to chapter 1.2.

Supply power to the evaluation board with the program written and connect between a PC and the evaluation board with a USB cable.

When they are connected normally, the connected "COM**" (** indicates the port number) is displayed in the pull-down list for COM selection in the Main Window's Connection. This means the status where COM communication between the PC and the board is established and a communication can start.

Connection			
СОМ		Clock	
Status	OffLineMode	Select the conr the pu	nected COM port from Ill-down menu.

Figure 2-22 COM selection list of Main Window's Connection

2.7.2 Starting USB Communication

When you select "COM**" from the COM selection list, the communication connecting process starts. The connecting process status is displayed in "Status". When the connecting process is normally completed, "Connect – USB Serial Port" will be displayed in "Status". In this state, tools are enabled.

Connection						
COM	COM3	▼	Clock	Dis	plavs establishm	ent of
Status	Connect USB Se	erial Device	-		connection.	

Figure 2-23 Connection status view after connection establishment



If an error message is displayed when you select "COM**" from the COM selection list, the causes could be as follows; the USB connection may have a problem, the target board may not be turned on, or the communication clock setting may be wrong.

	×
Error	
[Message] Failed to COM connection. [Solution] Please reconnect after confirmation of connection cable, a board, a power supply and Frequency Clock setting. [Message Code] 1-1-31	~
OK	

Figure 2-24 Error message under COM connection



2.7.3 When using Communication Board for Tools (setting clock frequency)

When using a communication board for tools (MC-COM by Renesas or W2002 by Desk Top Laboratories), specify the communication speed by the following operation.

Click the "Clock" button in Main Window's "Connection" to display the Clock Setting dialog. The Clock Setting dialog displays the clock frequency that is currently set, so you can change this value. The changed clock frequency will take effect at the time of COM connection.



Figure 2-25 Clock Setting dialog

To set the clock frequency, obtain the value by multiplying the communication rate by 8.

Example:

When the communication rate is 1 Mbps, set the clock frequency to 8 MHz (8,000,000 Hz). When the communication rate is 5 Mbps, set the clock frequency to 40 MHz (40,000,000 Hz).

When using a MC-COM (Renesas communication board for tools), you can select the following communication rates (clock frequencies). Set it based on the jumper (JP2) of the MC-COM.

Table 2-2 Settings of MC-COM JP2 and selectable clock frequence	es
---	----

JP2	Selectable clock frequency
Short	1 Mbps (8 MHz), 5 Mbps (40 MHz), 7.5 Mbps (60 MHz), 10 Mbps (80 MHz),
	15 Mbps (120 MHz)
Open	6.25 Mbps (50 MHz), 8.33 Mbps (666666666 Hz), 12.5 Mbps (10 MHz),
	16.66 Mbps (133333333 Hz)





Figure 2-26 JP2 of MC-COM (communication board for tools)

Table 2-4 and 2-5 shows maximum communication rate on each MCU under conditions shown in this table.

Conditions

- Configure CPU clock frequency at maximum
- Applicable baud rate for MC-COM

Product name	CPU clock frequency	Supplied clock frequency	Baud rate	Sampling interval	Lib API(ics2_init) configured value	Lib API(ics2_init) configured value
		(UART)			4 th argument char speed	5 th argument char mode
RX72T	200MHz	50MHz	6.25Mbps	50us/4ch	0	2
RX66T	160MHz	40MHz	5Mbps	50us/4ch	0	2
RX26T	120MHz	60MHz	7.5Mbps	50us/4ch	0	2
RX24T/U	80MHz	40MHz	5Mbps	50us/4ch	0	2
RX23T	40MHz	40MHz	5Mbps	50us/4ch	0	2
RX13T	32MHz	32MHz	1Mbps	200us/4ch	3	2

Table 2-1	Maximum communication rate (for RX microcontroller)
-----------	---



Product	CPU clock	Supplied clock frequency	Baud rate	Sampling	Lib API(ics2_init) configured value	Lib API(ics2_init) configured value
		(UART)			3 rd argument char speed	4 th argument char mode
RA8T1	480MHz	120MHz	10Mbps	50us/4ch	1	2
RA6T3	200MHz	100MHz	16.6Mbps	50us/4ch	0	2
RA4T1	100MHz	100MHz	16.6Mbps	50us/4ch	0	2
RA6T2	240MHz	120MHz	10Mbps	50us/4ch	1	2
RA6T1	120MHz	120MHz	15Mbps	50us/4ch	0	2

 Table 2-2
 Maximum communication rate (for RA microcontroller)

2.7.4 When using Built-in type Communication Library (setting baud rate)

When using the built-in type communication library, set the baud rate by the following operation.

Select "Baud rate Dialog" from Main Window's "Option" menu to display the BaudrateSetting dialog. The dialog displays the baud rate that is currently set, so you can change this value. The changed baud rate will take effect at the time of COM connection.



Figure 2-27 Setting baud rate

Set the baud rate to the value that is set by the program including the built-in type communication library. In addition, the set value must be a value that can be set to the connected USB serial conversion board from the PC. You can check the configurable values in the properties of the COM port (port setting) from Device Manager of Windows control panel.



2.7.5 In Case of USB Connection Trouble

If any problem occurs on USB connection, solve it by the following operations.

- Check the connection to the evaluation board and the power supply to the board.
- If no connectable COM is displayed, check the authentication status. (Select "Load Authentication File" from "Help" menu and check the message on the authentication screen.)
- Confirm that the execution program of the evaluation board and the variable list loaded into Renesas Motor Workbench are both generated in the same build.
- Reset the evaluation board, reconnect the USB cable, or change the USB port to another one, etc.



2.8 Description of Operation

2.8.1 Launching Internal Tool

When a Map or RMT file is loaded and the target is connected normally, icons for available tools will be shown in Main Window's Select Tool. (The view of Main Window and available tools vary depending on the sample program.)

Click an icon in Select Tool to start the tool, and the window for the tool will open.

File Option	Help Displays target is co	when the nnected via SB.		Indicates the RMT file that are loaded	e and Map file currently.
Connection		File In	formation	/	
сом	COM3	Clock RMT Fi	le RA6T1_MRSSK2_SPM_LESS_FC	DC_TUNER_RV 4/5/2023 2:54:08 PN	и
splays only for the sample program ovided by Renesas.	Connect USB Serial Device	Map Fi	le -	•	-
Configuration	\mathbf{X}	Select	: Tool		
CPU	RA6T1				
Motor Type	Brushless DC Motor	1 - 1 - 1 D		Sele	ect a tool to st
Control	Software for Tuner(Speed contr	rol)			om Select Too
	RSSK for Motor		Easy Analyzer	Tuner	
Inverter	R35K for Motor				
Inverter Project File Path	D:\Work\RA6T1_MRSSK2_SPM_LESS_	_FOC_TUNER_RV200	▼ ¢	Details	•
Inverter Project File Path	D:\Work\RA6T1_MRSSK2_SPM_LESS	_FOC_TUNER_RV200	Date Modified	Details	•

Figure 2-28 Main Window (after the target is connected)



2.8.2 Basic GUI Operation

This section describes the basic GUI operations for Main Window and the internal tools (Easy, Analyzer, Servo, and Tuner) of Renesas Motor Workbench.

For details on each function, refer to the Renesas Motor Workbench User's Manual.

(1) Tool switch button

When you launch a tool from Select Tool on Main Window, "Tool switch button" will be displayed on the top of the screen of each tool. Click it to switch to another tool.

Note that you cannot switch tools during operating (driving a motor).

(2) Main Window switch button

"Main Window button" is shown on the upper right of the screen of each tool. Click it to go back to Main Window.

Note that you cannot switch to Main Window during operating (driving a motor).



Figure 2-29 Tool switch button and Main Window button



(3) Window view switch button

In the screen of each tool (Analyzer, Servo, Tuner), you can change the window view by clicking the following buttons. You can also release the maximized window.

You can select the frontmost window with the window list button (button (a) in the figure below). However, when a window is framed out of the tool, the window is not displayed in the list of active windows. (In this case, select the window from Windows Task Bar.)



Figure 2-30 Window view switch button in each tool window



(4) Framing Out/In Window

You can move (frame out) tool windows (except for some tools) outside of the tool frame by dragging the title. In reverse, you can return (frame in) the window to the original tool frame by dragging it.



Figure 2-31 Framing out and in window



2.9 Saving RMT file and terminating Renesas Motor Workbench

The RMT file can be saved from "Save" or "Save As" of Main Window's "File" menu.



Figure 2-32 Saving RMT file

To terminate Renesas Motor Workbench, select "Exit" from Main Window's "File" menu or press the close button on the upper right corner of the screen.

On termination, the exit screen is shown as below.

	×
Information	
[Message] Do you want to exit Renesas Motor Workbench after saving the set information to the RMT file? [Message Code] 1-2-16	~
	~
Save Save As No Save Cancel	

Figure 2-33 Renesas Motor Workbench exit screen

The functions of each button on the exit screen are as follows.

Table 2-3 Functions of buttons on exit screen

Button	Function
Save	Saves the environment information of Renesas Motor Workbench into the loaded RMT file by overwriting it and exits.
Save As	Saves the environment information of Renesas Motor Workbench into a new RMT file with a new name specified and exits.
No Save	Exits without saving the environment information of Renesas Motor Workbench.
Cancel	Cancel the termination process.



3. How to Generate Map File

When Renesas Motor Workbench loads the global variable information (Map file) of the program written into the MCU on the evaluation board, it gains access to the global variables. This Map file can be generated together with the execution file (control software) in the program building.

The followings are setting procedures to generate a Map file in the Integrated Development Environment "CS+ (CC compiler)", "e² studio", and "e² studio (RA)".

3.1 [RX] In CS+ (CC Compiler) Environment

Setting procedure:

- ① Launch CS+ and display the target project. Open the project information from "Project Tree".
- (2) Right-click on "CC-RX (Build Tool)" in "Project Tree" and select "Property". The "Property" window will be displayed on the right side of the screen.
- ③ Click "Link Options" in "Property".
- ④ Open "List" and "Others".
- ⑤ See "Outputs the linkage list file" under "List" and if it is set to "Yes (List contents=ALL)(-LISt -SHow=ALL)", skip to ⑨. If not, proceed to ⑥.
- 6 Select "Yes (List contents=specify) (-LISt)" to "Outputs the linkage list file" under "List".
- ⑦ Select "No" to "Outputs a symbol name list in a module" under "List".
- 8 Add "-Show=Symbol,struct" (comma-delimited) to "Other additional options" under "Others".
- 9 Build with the above settings, and a Map file will be generated including "variable information".



2 East View Project	Build Debug Tool Window Help	Prop	perty			+ ×
🕅 Start 🛛 🛃 🗐	メ 単語 ち の 器 単 単	🛛 🖓 🖓	RX Property			â P - +
	🔲 💭 🗔 🗇 🚺 Solution List	V List	nuts the unkage list file		Yes(List contents=specify(-LISt)	^
Project Tree		Outp	puts a symbol name list in a	module	No	
000	1 Right-click on CC-RX.	Outp	puts the number of symbol re puts the cross-reference infor	terences rmation	No	
PY72T MPSCK SPM	LESS FOR CSR PIVION /Pre	Show	ws the total sizes of sections		(Onen "list" and "Others"	
R5F572TKCxFB (MAR	(rocontroller)	Outp	puts vector information puts relocation attributes rela	ted to sections	Groupen List and Others.	
CC-RX (Build Too	Duild Devices 17	V Opt	timization		Na antigia (NOODigia)	
RX E2 Lite (Debug	Behald Project F7	> Sec	ction		No opumize(-NOOPumize)	
Program Analyzer	Con Rebuild Project Shirt+P/	Yes	NTS NTS			
- application	Clean Project		pute a stack use information f	file	No	
🕀 - 🔜 middle	Set to Default (2) Select "Property".	Com	npresses the debugging infor anges the warning-level and e	mation rror-level messages to information-le	vel me No	
driver	Import Build Options	Cha	inges the information-level an	d error-level messages to warning-le	vel me No	
ter-ter config	Set Link Order	Dele	etes local symbol name inform	nation	No	
F	Property	Disp	plays the total sizes of section plays the convright information	ns o	Yes/ Total aise)	
		> Corr	nmands executed before link	processing 3	Select "Link Options" tab.	
		> Com Othe	nmands executed after link pr er additional options	rocessing	-show=symbol,struct	
		> Com	nmand line		Command line[12]	~
1		Outputs	ts the linkage list file	list file		
		This cor	rresponds to the -list and -sho	ow option of the linker.		
		Comm	non Options 🖌 Compile Op	tions / AssembleOption / Lin	k Options / Hax Output Options / Library Generate Options /	•
				<u> </u>		
		¥				
Property		-	×			
CC-RX Property	1	_ م <u>د</u>	+			_
Input						
O • •			(5)	It it is set to "Yes	(-LIST-Show=ALL)", proceed to	
Output					(, , p	
List			bui	ld.	(, , p, , p, ,	
Cutout List Outputs the link Optimization	rage list file Yes(List contents=ALL)((-LISt -SHow=ALL)	bui	ld.	(, , p ;	
Cutput List Outputs the link Optimization Section	age list file Yes(List contents-ALL)((-LISt -SHow=ALL)	bui	ld.	Otherwise	
Cutout List Outputs the link Optimization Section Verify	cage list file Yes(List contents=ALL)((-LISt -SHow=ALL)	bui	ld.	Otherwise	
Outout List Outputs the link Optimization Section Verify	rage list file Yes(List contents=ALL)((-LISt -SHow=ALL)	bui	ld.	Otherwise	
Output List Outputs the link Optimization Section Verify	rage list file Yes(List contents=ALL)((-LISt -SHow=ALL)	bui	ld.	Otherwise	
 Outout List Outputs the link Optimization Section Verify 	rage list file Yes(List contents=ALL)((-LISt -SHow=ALL)	bui	ld.	Otherwise	
Cutout List Outputs the link Optimization Section Verify	rage list file Yes(List contents=ALL)	(-LISt -SHow-ALL)	bui	ld.	Otherwise	
List Outputs the link Optimization Section Verify	rage list file Yes(List contents=ALL)((-LISt -SHow=ALL)	bui	ld.	Otherwise	
List Outputs the link Optimization Section Verify	rage list file Yes(List contents=ALL)	(-LISt -SHow=ALL)	bui	ld.	Otherwise	"Outouto de la
List Outputs the link Optimization Section Verify	rage list file Yes(List contents-ALL)	(-LISt -SHow=ALL)	bui	© Select "Yes	Otherwise (List contents=specify) (-LISt)" to	, "Outputs the
Ustroat List Outputs the link Optimization Section Verify	tage list file Yes(List contents-ALL)((LISt -SHow=ALL)	bui	© Select "Yes linkage list file	Otherwise List contents=specify) (-LISt)" to	"Outputs the
Uitod List Optimization Section Verify	rage list file Yes(List contents=ALL)	(-LISt -SHow=ALL) Yes(List contents=specify)	(415)	6 Select "Yes linkage list file	Otherwise (List contents=specify) (-LISt)" to ".	"Outputs the
Utotot List Outputs the link Section Section Verify	tage list file Yes(List contents-ALL)(Property CCRX Property List Outputs the linkage list file Outputs a symbol name list in a module	(-LISt -SHow=ALL) Yes(List contents=specify) No	(LUS)	© Select "Yes linkage list file ⑦ Select "No"	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	"Outputs the in a module" unde
List Outputs the link Optimization Section Verify	Acage list file Yes(List contents-ALL) Property CCRX Property List Outputs a private file Outputs the inneher of sympol references Chards the number of sympol	(-LISt -SHow=ALL) Yes(Uit contents=specify) No No	(US)	© Select "Yes linkage list file ⑦ Select "No" ".ist"	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	, "Outputs the in a module" unde
Uist Outputs the link Optimization Section Verify	tage list file Yes(List contents=ALL) Property CCRX Property CCRX Property CCRX Property CLRX Property List as ymbol name list in a module Outputs the inkage list file Outputs the inkage list file Outputs the inkage list file Outputs the inkage list in a module Output the inkage list file Outputs	(-LISt -SHow=ALL) Yes(List contents=specify) No No No No	(LIS)	 Select "Yes linkage list file Select "No" "List". 	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	"Outputs the
Uist Outputs the link Optimization Section Verify	cage list file Yes(List contents-ALL)((LISt -SHow=ALL) Yes(List contents=specify) No No No No No No No No	(4JS)	© Select "Yes linkage list file ⑦ Select "No" "List".	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	"Outputs the in a module" unde
Uitod List Outputs the link Optimization Section Verify		(-LISt -SHow=ALL) Yes(List contents=specify) No No No No No No No No No	(415)	 Select "Yes linkage list file Select "No" "List". 	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	, "Outputs the in a module" unde
Uist Outputs the link Optimization Section Verify		(-LISt -SHow=ALL) Yes[List contents=specify] No No No No No No No No No	(LIS)	 Select "Yes linkage list file Select "No" "List". 	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	"Outputs the in a module" unde
Ustod Vist Outputs the link Optimization Section Verify	cage list file Yes(List contents-ALL) Property C-RX Property C-RX Property Viat Outputs the Intrage list file Outputs the Intrage list file Outputs the Intrage list file Outputs the Intrade list of Sections Outputs the Interse references Outputs	(LISt -SHow=ALL) Yes(Uit contents=specify) No No No No No No No No No No	(US)	© Select "Yes linkage list file ⑦ Select "No" "List".	Otherwise (List contents=specify) (-LISt)" tc ". to "Outputs a symbol name list	, "Outputs the in a module" unde
Uist Outputs the link Optimization Section Verify	cage list file Yes(List contents=ALL)((-LISt -SHow=ALL) Yes(List contents=specify) No No No No No No No No No No No No No	(US)	6 Select "Yes linkage list file D Select "No" "List".	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	o "Outputs the
Uist Outputs the link Optimization Section Verify	Acage list file Yes(List contents-ALL) Yes(List contents-ALL) Comparison of the second of the se	(LISt -SHow=ALL) Yes(List contents=specify) No No No No No No No No No No	(LIS)	© Select "Yes linkage list file ⑦ Select "No" "List".	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	"Outputs the in a module" unde
Uitod List Outputs the link Optimization Section Verify		(LISt -SHow=ALL) Yes(List contents=specify) No No No No No No No No No No No No No	(KLIS)	Select "Yes linkage list file Ø Select "No" "List".	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	, "Outputs the in a module" unde
Uist Outputs the link Optimization Section Verify		(-LISt -SHow=ALL) Yes(List contents=specify) No No No No No No No No No No No No No	(US)	 Select "Yes linkage list file Select "No" "List". 	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	"Outputs the
Utitod List Outputs the link Optimization Section Verify	CRRX Property ✓ CRRX Property ✓ CRRX Property ✓ CRRX Property ✓ Usit Outputs a spin-binaria list in a module Outputs the number of symbol references Outputs the spin-binaria list in a module Outputs a spin-binaria list in a module Outputs a spin-binaria Optimization Optimization Optimization Outputs a stack use information file Curputs a stack use information Outputs a stack use information Outputs a stack use information Charges the warming-level and area-level messages to information	(LISt -SHow=ALL) Yes(Uit contents=specify) No No No No No No No No No No	(LUS)	© Select "Yes linkage list file ⑦ Select "No" "List".	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	, "Outputs the in a module" unde
Uist Outputs the link Optimization Section Verify		(LISt -SHow=ALL) Yes(List contents=specify) No	(4.15)	6 Select "Yes linkage list file 7 Select "No" "List".	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	o "Outputs the
Uist Outputs the link Outputs that Section Verify	cage list file Yes(List contents-ALL)(Yes(List contens-ALL)(Yes(List contens-ALL)(Yes(Lis	(-LISt -SHow=ALL) Yes[List contents=specify] No No No No No No No No No No	(LUS)	© Select "Yes linkage list file ⑦ Select "No" "List".	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	"Outputs the in a module" unde
Uitod List Outputs the link Optimization Section Verify		(-LISt -SHow=ALL) Yee(List contents=specify) No	(KUS)	Select "Yes linkage list file Ø Select "No" "List".	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list	, "Outputs the in a module" unde
Uist Outputs the link Optimization Section Verify		(-LISt -SHow=ALL) Yes(List contents=specify) No	(US)	 Select "Yes linkage list file Select "No" "List". 	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list /=Symbol,struct" (comma-delimi	o "Outputs the in a module" unde
Uist Outputs the link Optimization Section Verify	cage list file Yes(List contents-ALL)	(-LISt -SHow=ALL) Yest[List contents=specify] No	(LUS)	 Select "Yes linkage list file Select "No" "List". 	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list /= Symbol,struct" (comma-delimi	, "Outputs the in a module" unde
Uist Outputs the link Optimization Section Verify		(LISt -SHow=ALL) Yes(List contents=specify) No	(415)	 Select "Yes linkage list file Select "No" "List". Add "-Show "Other addition 	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list v=Symbol,struct" (comma-delimi nal options" under "Other".	o "Outputs the in a module" unde
Uitod List Outputs the link Outputs the link Section Section Verify	cage list file Yes(List contents-ALL)(Yes(List contents-ALL)) Yes(List contents-ALL) Yes(List contents-ALL	(LISt -SHow=ALL) Yes[List contents=specify] No	(LIS)	 Select "Yes linkage list file Select "No" "List". 8 Add "-Show "Other addition 	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list v=Symbol,struct" (comma-delimi nal options" under "Other".	"Outputs the in a module" unde ted) to
Uitod List Outputs the link Optimization Section Verify	cage list file Yes(List contents-ALL)(Yes(List contents-ALL)) Property CCRX Property CCRX Property CCRX Property Yes Compute the linkage list file Outputs the constraints Outputs Outputs Output Output Output Outputs Output O	(-LISt -SHow=ALL) Yest[List contents=specify] No	(KLIS)	 Select "Yes linkage list file Select "No" "List". B Add "-Show "Other additio 	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list /=Symbol,struct" (comma-delimi nal options" under "Other".) "Outputs the in a module" unde
Uitod List Outputs the link Optimization Section Verify	cage list file Yes(List contents=ALL)(Verify Property CCRX Property CCRX Property CCRX Property CCRX Property CCRX Property CCRX Property CLA CCRX Property CCRX Property CLA C	(LISt -SHow=ALL) Yes(List contents=specify) No	(-US)	 Select "Yes linkage list file Select "No" "List". Add "-Show "Other additio 	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list /=Symbol,struct" (comma-delimi nal options" under "Other".	"Outputs the in a module" unde
Uitod List Outputs the link Optimization Section Verify	cage list file Yes(List contents-ALL) Yes(List contens-ALL) Yes(List contents-ALL)	(LISt -SHow=ALL) Yest[List contents=specify] No	(LLS)	 Select "Yes linkage list file Select "No" "List". 8 Add "-Show "Other addition 	Otherwise (List contents=specify) (-LISt)" to ". to "Outputs a symbol name list /= Symbol,struct" (comma-delimi nal options" under "Other".	, "Outputs the in a module" unde ted) to

Figure 3-1 CS+ (CC Compiler) property setting



Tips : Copy a Map file to the location where a RMT file is saved after the build.

When you load a Map file after loading a RMT file, the dialog box is opened with the directory path of the loaded RMT file specified. This is because the directory is opened that has been the most recently operated in the file selection dialog box. Therefore, if you have copied a Map file to a RMT file directory in advance, you can select the RMT/Map file without changing the directory path in the dialog box.

In [Link Options] > [Others] > [Command executed after the build], enter the command as below to copy the Map file to the directory where the RMT file is saved, and the Map file will be copied automatically after the build.

copy %BuildModeName%\%ProjectName%.map app\rmw\%ProjectName%.map

	Changes the warning-level and error-level messages to information-leve	l No
(Changes the information-level and error-level messages to warning-leve	l No
(Changes the information-level and warning-level messages to error-leve	l No
1	Deletes local symbol name information	No
1	Displays the total sizes of sections	Yes(-Total_size)
1	Displays the copyright information	No(-NOLOgo)
> (Commands executed before link processing	Commands executed before link processing[0]
\sim	Commands executed after link processing	Commands executed after link processing[1]
	[0]	copy %BuildModeName%¥%ProjectName%.map lib¥ics¥%ProjectName%.map
(Other additional options	-show=symbol,struct
> (Command line	Command line[11]

Figure 3-2 Settings for copying Map file in CS+ (CC Compiler)

3.2 [RX] In e² studio Environment

Setting procedure:

- (1) Launch e^2 studio and display the target project.
- (2) Right-click on the target project from "Project Explorer" and select "Properties". The "Properties" window will be displayed.
- ③ Open "C/C++ Build" on the left and select "Settings".
- ④ Select "Tool Settings" tab on the right from "Settings".
- 5 Select "User" under "Linker".
- In "User-defined options (add after command line)" on the right bottom, press "+" button and add "-Show=Symbol,struct" (comma-delimited)
- \bigcirc Build with the above settings, and a Map file will be generated including "variable information".





Figure 3-3 e² studio property setting



Tips: Copy a Map file to the location where a RMT file is saved.

When you load a Map file after loading a RMT file, the dialog box is opened with the directory path of the loaded RMT file specified. This is because the directory is opened that has been the most recently operated in the file selection dialog box. Therefore, if you have copied a Map file to a RMT file directory in advance, you can select the RMT/Map file without changing the directory path in the dialog box.

In [C/C++Build] > [Settings] > [Build Step] tab > [Post-build steps], enter the command as below to copy the Map file to the directory where the RMT file is saved, and the Map file will be copied automatically after the build.

copy \${ProjName}.map ..\app\rmw\\${ProjName}.map

DefaultBuild [Active]
js Toolchain Device 🎤 Build Steps 🏆 Build Artifact 🗟 Binary Parsers
eps):
teps):
Name}.map¥app¥rmw¥\${ProjName}.map

Figure 3-4 Settings for copying Map file in e2studio



3.3 [RA] In e² studio Environment

Setting procedure:

- ① Copy "ElfMapConverter.exe" attached in Renesas Motor Workbench to the output folder of the project.
- 2 Create a bat file that describes the following command. (The file can be named as any name.)

arm-none-eabi-objdump.exe -W project-name.elf>project-name.txt ElfMapConverter.exe project-name.txt

- ③ Launch "e² studio" and display the project.
- ④ Right-click on the target project from "Project Explorer" and select "Properties". The "Properties" window will be displayed.
- (5) Open "C/C++ Build" on the left and click "Settings".
- 6 Select "Build Steps" tab on the right in "Settings".
- ⑦ Add the bat file created in ② to the command field of "Post-build steps". (The file is named "map.bat" for example here.)
- 8 Build with the above settings, and a Map file (.rmap) will be generated including the "variable information".

type filter text	Settings	← ▼ ⇒ ₹ 8
 > Resource Builders ♥ C/C++ Build Build Variables Environment Logging ▶ Settings Tool Chain Editor > C/C++ General Project Natures Project References > Renesas QE Run/Debug Settings 	Configuration: Debug [Active]	 Manage Configurations uild.gnuarm.config.elf.debug.1116809476 sers
(?)		Apply and Close Cancel

Figure 3-5 e² studio (RA) property setting



Tips: Copy a Map file to the location where a RMT file is saved.

When you load a Map file after loading a RMT file, the dialog box is opened with the directory path of the loaded RMT file specified. This is because the directory is opened that has been the most recently operated in the file selection dialog box. Therefore, if you have copied a Map file to a RMT file directory in advance, you can select the RMT/Map file without changing the directory path in the dialog box.

In addition, if you have modified the extension of the Map file generated in this section from *.rmap to *.map and copy the file, you do not need to change the file type in the dialog box for loading Map files. If you add the command as below in the bat file described in the above step ②, the Map file whose extension has been modified will be copied automatically after the build. In the following example,

"_conv" is added to the copied Map file name so that you can identify the extension changed from *.rmap to *.map.

copy ProjectName.rmap <RMT file save directory>\ProjectName_conv.map



4. Communication Library

4.1 Communication Library for each MCU

To use Renesas Motor Workbench, it is necessary to include the communication library into a user program. The communication library is provided in the package of Renesas Motor Workbench (in "communication library" folder). Table 4-1 shows the communication libraries provided for each MCU.

Supported CPU	RX23T RX24T RX24U				
File	ics_RX23T.obj	ics_RX24T.obj	ics_RX24U.obj		
	ics_RX23T.h	ics_RX24T.h	ics_RX24U.h		
Communication rate	0.5 Mbps to 5 Mbps				
Port	SCI1 TXD1:PD3	SCI1 TXD1:PD3	SCI1 TXD1:PD3		
	RXD1:PD5	RXD1:PD5	RXD1:PD5		
	SCI5 TXD5:PB5	SCI5 TXD5:PB5	SCI5 TXD5:PB5		
	RXD5:PB6	RXD5:PB6	RXD5:PB6		
	SCI5 TXD5:PB2	SCI6 TXD6:PB2	SCI6 TXD6:PB2		
	RXD5:PB1	RXD6:PB1	RXD6:PB1		
		SCI6 TXD6:PB0	SCI6 TXD6:PB0		
		RXD6:PA5	RXD6:PA5		
		SCI6 TXD6:P81	SCI6 TXD6:P81		
		RXD6:P80	RXD6:P80		
Supported variable type	8bit unsigned integer (number display/setting/waveform display)				
	8bit signed integer (num	per display/setting/wavefor	m display)		
	16bit unsigned integer (n	umber display/setting/wav	eform display)		
	16bit signed integer (nun	nber display/setting/wavefo	orm display)		
	32bit unsigned integer (n	umber display/setting/wav	eform display)		
	32bit signed integer (nun	nber display/setting/wavefo	orm display)		
	32bit IEEE754 floating point (number display/setting/waveform display)				
CPU resource to use	SCIx RX, SCIx TX, DTC				

Table 4-1 Communication library (for RX) 1/3



Supported CPU	RX66T	RX72T	RX72M		
File	ICS2_RX66T.lib	ICS2_RX72T.lib	ICS2_RX72M.lib		
	ICS2_RX66T.h	ICS2_RX72T.h	ICS2_RX72M.h		
Communication rate	0.5 Mbps to 7.5 Mbps	•			
Port	SCI1 TXD1:PD3	SCI5 TXD5:PB5	SCI1 TXD5:PF0		
	RXD1:PD5	RXD5:PB6	RXD5:PF2		
	SCI5 TXD5:PB5	SCI6 TXD6:PB0	SCI2 TXD6:P50		
	RXD5:PB6	RXD6:PB1	RXD6:P52		
	SCI6 TXD6:PB0	SCI8 TXD8:PC1	SCI3 TXD8:P23		
	RXD6:PB1	RXD8:PC0	RXD8:P25		
		SCI12 TXD12:PB5	SCI4 TXD8:PB1		
		RXD12:PB6	RXD8:PB0		
			SCI5 TXD5:PA4		
			RXD5:PA3		
	SCI6 TXD6:P00				
		RXD6:P01			
			SCI6 TXD8:PB1		
			RXD8:PB0		
			SCI8 TXD12:PJ2		
			RXD12:PC6		
			SCI8 TXD12:PJ2		
			RXD12:PJ1		
Supported variable type	8bit unsigned integer (number display/setting/waveform display)				
	8bit signed integer (numb	er display/setting/waveforr	n display)		
	16bit unsigned integer (number display/setting/waveform display)				
	16bit signed integer (num	ber display/setting/wavefo	rm display)		
	32bit unsigned integer (n	umber display/setting/wave	form display)		
	32bit signed integer (num	ber display/setting/wavefor	rm display)		
	32bit IEEE754 floating point (number display/setting/waveform display)				
CPU resource to use	SCIx RX, SCIx TX, DTC				

Table 4-2 Communication library (for RX) 2/3



Supported CPU	RX13T	RX26T	
File	ICS2_RX13T.lib	ICS2_RX26T.lib	
	ICS2_RX13T.h	ICS2_RX26T.h	
Communication rate	0.5 Mbps to 4 Mbps	0.5 Mbps \sim 7.5 Mbps	
Port	SCI1 TXD1:PD3	SCI1 TXD1:PD3	
	RXD1:PD5	RXD1:PD5	
	SCI1 TXD1:PB6	SCI5 TXD5:PD7	
	RXD1:PB7	RXD5:PE0	
	SCI5 TXD5:PB6	SCI5 TXD5:PB5	
	RXD5:PB7	RXD5:PB6	
	SCI5 TXD5:PB2	SCI6 TXD6:P81	
	RXD5:PB1	RXD6:P80	
	SCI5 TXD5:P23	SCI6 TXD6:PB2	
	RXD5:P24	RXD6:PB1	
	SCI12 TXD12:PB0	SCI12 TXD12:PD4	
	RXD12:P94	RXD12:PD6	
		SCI12 TXD12:P01	
		RXD12:P00	
		SCI12 TXD12:P81	
		RXD12:P80	
		SCI12 TXD12:P23	
		RXD12:P22	
		SCI12 TXD12:PB5	
		RXD12:PB6	
Supported variable type	8bit unsigned integer (number displ	ay/setting/waveform display)	
	8bit signed integer (number display/	/setting/waveform display)	
	16bit unsigned integer (number display/setting/waveform display)		
	16bit signed integer (number display/setting/waveform display)		
	32bit unsigned integer (number display/setting/waveform display)		
	32bit signed integer (number display	y/setting/waveform display)	
	32bit IEEE754 floating point (number display/setting/waveform display		
CPU resource to use	SCIx RX, SCIx TX, DTC		

Table 4-3 Communication library (for RX) 3/3



Supported CPU	RL78/G14	RL78/G1F	
File	ics2_RL78G14.Lib	ics2_RL78G1F.Lib	
	ics2_RL78G14.h	ics2_RL78G1F.h	
Communication	0.5 Mbps to 5.33 Mbps		
rate			
Port	SCI0 TXD0:P51, RXD0:P50	SCI0 TXD0:P51, RXD0:P50	
	SCI0 TXD0:P12, RXD0:P11	SCI0 TXD0:P17, RXD0:P16	
	SCI0 TXD0:P17, RXD0:P16	SCI1 TXD0:P00, RXD0:P01	
	SCI1 TXD0:P00, RXD0:P01	SCI1 TXD0:P02, RXD0:P03	
	SCI1 TXD0:P02, RXD0:P03	SCI1 TXD0:P72, RXD0:P73	
	SCI1 TXD0:P72, RXD0:P73	SCI1 TXD0:P77, RXD0:P76	
	SCI1 TXD0:P77, RXD0:P76	SCI2 TXD2:P13, RXD2:P14	
	SCI1 TXD0:P82, RXD0:P81		
	SCI2 TXD2:P13, RXD2:P14		
	SCI2 TXD0:P77, RXD0:P76		
	SCI3 TXD0:P144, RXD0:P143		
Supported	8bit unsigned integer (number display/setting/waveform display)		
variable type	8bit signed integer (number display/setting/waveform display)		
	16bit unsigned integer (number display/setting/waveform display)		
	16bit signed integer (number display	/setting/waveform display)	
	32bit unsigned integer (number disp	lay/setting)	
	32bit signed integer (number display	/setting)	
	8bit bool (number display/setting)		
	8bit logic (number display/setting)		
CPU resource to	SCIx RX、SCIx TX、DTC		
use			

Table 4-4 Communication library (for RL) 1/2



Supported CPU	RL78/G24		
File	ics2_RL78G24.Lib		
	ics2_RL78G24.h		
Communication rate	0.5 Mbps to 8 Mbps		
Port	SCI0 TXD0:P51, RXD0:P50		
	SCI0 TXD0:P12, RXD0:P11		
	SCI0 TXD0:P17, RXD0:P16		
	SCI1 TXD0:P00, RXD0:P01		
	SCI1 TXD0:P02, RXD0:P03		
	SCI1 TXD0:P30, RXD0:P31		
	SCI1 TXD0:P72, RXD0:P73		
	SCI2 TXD2:P10, RXD2:P11		
	SCI2 TXD2:P13, RXD2:P14		
	SCI2 TXD0:P77, RXD0:P76		
Supported variable	8bit unsigned integer (number display/setting/waveform display)		
type	8bit signed integer (number display/setting/waveform display)		
	16bit unsigned integer (number display/setting/waveform display)		
	16bit signed integer (number display/setting/waveform display)		
	32bit unsigned integer (number display/setting)		
	32bit signed integer (number display/setting)		
	8bit bool (number display/setting)		
	8bit logic (number display/setting)		
CPU resource to use	SCIx RX、SCIx TX、DTC		

Table 4-5 Communication library (for RL) 2/2



Supported CPU	RA6T1	RA6T2	
File	ICS2_RA6T1.o	ICS2_RA6T2.o	
	ICS2_RA6T1.h	ICS2_RA6T2.h	
Communication rate	0.5 Mbps to 15.0 Mbps	0.5 Mbps to 20.0 Mbps	
Port	SCI0 TXD0:P101	SCI9 TXD9:PD05	
	RXD0:P100	RXD9:PD06	
	SCI4 TXD4:P205		
	RXD4:P206		
	SCI9 TXD9:P109		
	RXD9:P110		
Supported variable	8bit unsigned integer (number display/setting/waveform display)		
type	8bit signed integer (number display/setting/waveform display)		
	16bit unsigned integer (number display/setting/waveform display)		
	16bit signed integer (number display/setting/waveform display)		
	32bit unsigned integer (number display/setting/waveform display)		
	32bit signed integer (number display/setting/waveform display)		
	32bit IEEE754 floating point (number display/setting/waveform		
	display)		
CPU resource to use	SCIx RX, SCIx TX	SCIx RX, SCIx TX	

Table 4-6 Communication library (for RA) 1/2

Table 4-7 Communication library (for RA) 2/2

Supported CPU	RA6T3 RA4T1 RA8T1			
File	ICS2_RA6T3.o	ICS2_RA4T1.o	ICS2_RA8T1.o	
	ICS2_RA6T3.h	ICS2_RA4T1.h	ICS2_RA8T1.h	
Communication rate	$0.5~{ m Mbps} \sim 16.67$	$0.5~{ m Mbps} \sim 16.67$	0.5 Mbps \sim 20.0 Mbps	
	Mbps	Mbps		
Port	SCI0 TXD0:P101	SCI0 TXD0:P101	SCI1 TXD1:P213	
	RXD0:P100	RXD0:P100	RXD1:P212	
	SCI0 TXD0:P213	SCI0 TXD0:P213	SCI2 TXD2:PA03	
	RXD0:P212	RXD0:P212	RXD2:PA02	
	SCI0 TXD0:P411	SCI0 TXD0:P411	SCI3 TXD3:P409	
	RXD0:P410	RXD0:P410	RXD3:P408	
	SCI9 TXD9:P110	SCI9 TXD9:P110	SCI4 TXD4:P714	
	RXD9:P109 RXD9:P109		RXD4:P715	
Supported variable	8bit unsigned integer (number display/setting/waveform display)			
type	8bit signed integer (number display/setting/waveform display)			
	16bit unsigned integer (number display/setting/waveform display)			
	16bit signed integer (num	nber display/setting/wavefc	orm display)	
	32bit unsigned integer (number display/setting/waveform display)			
	32bit signed integer (num	nber display/setting/wavefc	orm display)	
	32bit IEEE754 floating point (number display/setting/waveform display)			
CPU resource to use	SCIx RX, SCIx TX	SCIx RX, SCIx TX	SCIx RX, SCIx TX	



4.2 Built-in Type Communication Library

The RA6T2, RA6T3 ,RA4T1 and RA8T1 supports built-in type communication libraries. When using the builtin type communication library, you can use a commercially available USB serial conversion module (isolated type).

The built-in type communication library is included in the package of Renesas Motor Workbench (in "RA6T2", "RA6T3", "RA4T1" and "RA8T1" folder in "communication library").

Occurrents of ODU	DAGTO	B A ATA	DA (T)	
Supported CPU	RA612	RA613	RA411	
File	ICS2_RA6T2_Built_in.o	ICS2_RA6T3_Built_in.o	ICS2_RA4T1_Built_in.o	
Communication rate	-	-	-	
Port	SCI9 TXD9:PD05	SCI0 TXD0:P101	SCI0 TXD0:P101	
	RXD9:PD06	RXD0:P100	RXD0:P100	
		SCI0 TXD0:P213	SCI0 TXD0:P213	
		RXD0:P212	RXD0:P212	
		SCI0 TXD0:P411	SCI0 TXD0:P411	
		RXD0:P410	RXD0:P410	
		SCI9 TXD9:P110	SCI9 TXD9:P110	
		RXD9:P109	RXD9:P109	
Supported variable type	8bit unsigned integer (number display/setting/waveform display)			
	8bit signed integer (number display/setting/waveform display)			
	16bit unsigned integer (nu	umber display/setting/wave	form display)	
	16bit signed integer (number display/setting/waveform display)			
	32bit unsigned integer (number display/setting/waveform display)			
	32bit signed integer (number display/setting/waveform display)			
	32bit IEEE754 floating point (number display/setting/waveform display)			
CPU resource to use	SCIx RX, SCIx TX	SCIx RX, SCIx TX	SCIx RX, SCIx TX	

Table 4-8 Built-in	type communication	library (for RA) 1/2
--------------------	--------------------	----------------------

Table 4-9 Built-in type communication library (for RA) 2/2

Supported CPU	RA8T1
File	ICS2_RA8T1_Built_in.o
Communication rate	-
Port	SCI1 TXD1:P213
	RXD1:P212
	SCI2 TXD2:PA03
	RXD2:PA02
	SCI3 TXD3:P409
	RXD3:P408
	SCI4 TXD4:P714
	RXD4:P715
Supported variable type	8bit unsigned integer (number display/setting/waveform display)
	8bit signed integer (number display/setting/waveform display)
	16bit unsigned integer (number display/setting/waveform display)
	16bit signed integer (number display/setting/waveform display)
	32bit unsigned integer (number display/setting/waveform display)
	32bit signed integer (number display/setting/waveform display)
	32bit IEEE754 floating point (number display/setting/waveform display)
CPU resource to use	SCIx RX, SCIx TX



4.3 How to Set User Program

4.3.1 DTC

Renesas Motor Workbench operates on a DTC (standard address mode), excluding RA MCUs. Therefore, it is necessary to define the DTC table in a user program.

Reserve the area for the DTC table within a program (within a file to call initialization functions described later).

- For RX microcontrollers Assign the DTC table to an address in the RAM so that the lower 12bit becomes 0.
- For RL microcontrollers Assign the DTC table to an address so that the lower 8bit becomes 0.

For code sample, refer to 4.3.4 How to Use Library Functions

When using an emulator such as E2, make sure that the user RAM area and the DTC table area do not overlap.

4.3.2 Interruption

Specify the functions (ics_int_sci_eri(), ics_int_sci_rxi()) for the user program interrupt vectors as shown in the code sample below (except for RA MCUs).

When using a project generated by the Renesas standard compiler with RX MCUs, specify them within intprg.c.

void Excep_SCI1_ERI1 (void){ ics_int_sci_eri(); }
void Excep_SCI1_RXI1 (void){ ics_int_sci_rxi(); }
void Excep_SCI5_ERI5 (void){ ics_int_sci_eri(); }
void Excep_SCI5_RXI5 (void){ ics_int_sci_rxi(); }

Figure 4-1 Code sample for communication interruption functions



4.3.3 Specification of Library Functions

In the communication library, the following two library functions are provided.

- void ics2_init(void* addr, char port, char level, char speed, char mode)
- void ics2_watchpoint(void)

The specifications of each function are shown below.

Table 4	I-10 Lib	rarv fun	ction (ic	s2 init) (1/2)
					, ,

Function name	void ics2_init(void* addr, char port, char level, char speed, char mode)		
Return value	void	None	
Argument	void* addr	The start address of a DTC vector table to be used: Users must reserve DTC vector table before calling this function. *No settings of this argument for RA series	
	char port	Set SCI port number and pins used for SCI.	
	char level	Set SCI interrupt level:	
		Set the appropriate interrupt level for the system, since appx.10 µsec interrupt occurs within minimum 2 msec intervals.	
		*No settings of this argument for RA series	
	char speed	Set communication rate:	
		Set the value of the communication rate by the following expression.	
		 Communication rate = CLOCK/ (x × (speed + 1)) [Mbps] 	
		CLOCK: The frequency of clock source supplied to SCI [MHz]	
		*For details, see the MCU's hardware manual.	
		— x: The fixed value for each MCU	
		For RX, RA6T1:x = 8	
		For RL78:x = 2	
		For RA6T2, RA4T1, RA6T3, RA8T1:x = 6	
		Ex.) When setting 1.0 Mbps for RX23T CLOCK = PCLKB = 40 [MHz], x = 8	
		1.0Mbps = 40MHz / (8 * (speed + 1) -> speed : 4	



Table 4-11 Library function (ics2_init) (2/2)

	char mode	mode 1 (32bits 8channels Action of 2 times transfer)
		In this mode, when the function "ics2_watchpoint()" is called once, 4 channels
		data are transferred with extraction of specified 8 channels data as 8bits,
		16bits, and 32bits for wave form display.
		At next time when "ics2_watchpoint()" is called, other remained 4 channels
		data are transferred without new extraction.
		That is to say, in case of "32bits 8channels mode", 8 channels data are
		transferred with 2 times call of "ics2_watchpoint()".
		• mode 2 (32bits 4channels Action of 1 time transfer)
		In this mode, when the function "ics2_watchpoint()" is called, 4 channels data
		are transferred with extraction of specified 4 channels data as 8bits, 16bits,
		and 32bits for wave form display.
		That is to say, in case of "32bits 4channels mode", 4 channels data are
		transferred with each call of "ics2_watchpoint()".
		It is impossible to display wave form over 5 channels.
		• mode 3 (32bits 12channels Action of 3 times transfer)
		In this mode, when the function "ics2_watchpoint()" is called once, 4 channels
		data are transferred with extraction of specified 12 channels data as 8bits,
		16bits, and 32bits for wave form display.
		At next time when "ics2_watchpoint()" is called, remained next 4 channels data
		are transferred without new extraction.
		And at 3rd time of call "ics2_watchpoint()", last remained 4 channels data are
		transferred.
		I hat is to say, in case of "32bits 12channels mode", 12 channels data are
		transferred with 3 times call of "ics2_watchpoint()".
Function	Initialization pr	ocess

Table 4-12 Library function (ics2_watchpoint)

Function	void ics2_watchpoint(void)				
name					
Return	void	None			
value					
Argumen	void	None			
t					
Function	Data transfer:				
	When the communication rate is BR [Mbps], the minimum sampling cycle is 70 + (180/BR) [µsec], and therefore the data transfer function is called at intervals greater than this value. For example, 250 [µsec] intervals or greater are required for communication rate at 1.0				
	[Mbps].				
	Note that when using a MC-COM (communication board for tools), the minimum sampling cycle is 10 + (180/BR) [µsec].				



4.3.4 How to Use Library Functions

(1) Calling on RX microcontroller

(a) Calling initialization function "ics2_init ()"
 Add the initialization function "ics2_init" to the initialization process part of a program as shown in the sample code below. Specify the argument based on the following rules.

The first argument:specify the start address (0 for the lower 12 bit) of a DTC tableThe second argument:select the port used from "ics_****.h" (see Table 4-1)The third argument:specify the tool interrupt levelThe forth argument:specify the communication rateThe fifth argument:specify the transfer mode

#include "ics_RX23T.h"

#pragma section DTCTBL
unsigned long dtc_table[256]; /* caution alignment 0x000 */
#pragma section
void main(void)
{
 ics2_init((void*) dtc_table, port, level, speed, mode);
}

Figure 4-2 Calling initialization function ics2_init() (RX23T sample code)

(b) Calling the data transfer function "ics2_watchpoint()"

Call the data transfer function "ics2_watchpoint()" at intervals longer than the minimum sampling cycle as shown in the sample code below.

```
void int_TM100u(void)  /* 100 µsec interval */
{
     if (3 <= g_u1_cnt_decimation) /* decimation of ICS call */
     {
        g_u1_cnt_decimation = 0;
        ics2_watchpoint();  /* data transfer */
     }
     g_u1_cnt_decimation++;
}</pre>
```





(2) Calling on RL microcontroller

(a) Calling initialization function "ics2_init()"

Add the initialization function" ics2_init" to the initialization process part of a program as shown in the sample code below. Specify the argument based on the following rules.

The first argument:specify the start address (0 for the lower 12 bit) of a DTC tableThe second argument:select the port used from "ics_****.h" (see Table 4-2)The third argument:specify the tool interrupt levelThe forth argument:specify the communication rateThe fifth argument:specify the transfer mode

#include "ics_RL78G1F_ca.h"

}

#pragma address dtc_tbl = 0xFFE00
char dtc_tbl[0xD0];
void main (void)
{
 ics2 init((void*) dtc table, port, level, speed, mode);



(b) Calling data transfer function "ics2 watchpoint()"

Call the data transfer function "ics2_watchpoint ()" at intervals longer than the minimum sampling cycle as shown in the sample code below.

__interrupt void int_TM50u(void) /* 50 µsec */
{
 if (4 <= g_u1_cnt_decimation) /* decimation of ICS call */
 {
 g_u1_cnt_decimation = 0;
 ics2_watchpoint(); /* data transfer */
 }
 g_u1_cnt_decimation++;
}</pre>





(3) Calling on RA microcontroller

(a) Calling initialization function "ics2_init()"

Add the initialization function" ics2_init()" to the initialization process part of a program as shown in the sample code below. Specify the argument based on the following rules.

The first argument:select the port used from "ics_****.h" (see Table 4-3)The second argument:specify the communication rate.The third argument:specify the transfer mode.



(b) Calling data transfer function "ics2_watchpoint()"
 Call the data transfer function "ics2_watchpoint ()" at intervals longer than the minimum sampling cycle as shown in the sample code below.

Figure 4-7 Calling data transfer function ics2_watchpoint() (RA6T1 sample code)



Revision History

		Revision	
Rev.	Date	Page	Contents
1.00	May.30.23	-	Issued as Quick Start Guide
1.01	Oct.27.23	-	Fixed 2.5.3 Acquiring New Authentication File
			Changed the URL for obtaining the authentication file to the RMW website.
1.02	Jan.30.24	-	Supported RA8T1
1.10	Feb.07.25	-	Fixed 1.1 Added Servo feature
			Fixed 1.2.2 System Requirement: Revised disk capacity, OS
			Fixed 1.3: Updated RL78 MCU table
			Fixed 2.2: Updated RL78 communication library
			Fixed 2.7.3: Added maximum communication rate table
			Fixed 4.1: Updated communication library for each MCU
			Fixed 4.3.3: Updated specification of library functions
1.11	Jun.30.25	6	1.4 Updated Supportive Information

Ref: Revision history of old version of User's Manual (R21UZ0004)

		Revision	
Rev.	Date	Page	Contents
1.00	Apr.05.17	-	Released version 1.0
2.00	Nov.27.18	-	Released version 2.0
2.01	May.16.19	-	Supported RX72T
2.02	Nov.29.19	-	Supported RX13T
2.03	Oct.28.20	-	Supported RA6T1
3.00	Dec.09.21	-	Released version 3.0
-	May.30.23	-	Changed to Quick Start Guide

Remarks

- This document is issued as "Quick Start Guide" by re-editing "Motor Control Development Support Tool Renesas Motor Workbench User's Manual (R21UZ0004)".
- In addition, these two existing Function Manuals (Application Notes)
 - Renesas Motor Workbench Functional Description (R20AN0527)
 - Renesas Motor Workbench Tuner Functional Description (R20AN0528) are re-edited into:
 - Renesas Motor Workbench User's Manual (R21UZ0004)



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 is supplied until the 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 systemevaluation 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.
- 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.
- 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.
- This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
 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/.