RA Family

Sensorless vector control for dual permanent magnetic synchronous motor

For Evaluation System for BLDC Motor Series

Introduction

This application note describes the sample program for a permanent magnetic synchronous motor drive with sensorless vector control based on Renesas microcontroller. This application note also describes how to use the motor control development support tool, 'Renesas Motor Workbench'.

The targeted software for this application is only to be used as reference purposes and Renesas Electronics Corporation does not guarantee the operations. Please use this after carrying out a thorough evaluation in a suitable environment.

Target Device

Operations of the target software of this application note are checked by using the following device.

- RA6T1 (R7FA6T1AD3CFP)

Contents

1. Overview

This application note explains how to implement the sensorless vector control software that drives dual permanent magnetic synchronous motor (PMSM) using the microcontroller RA series. The sample program of this application note can drive a motor easily with the kit of motor control (Renesas Flexible Motor Control series). And the program also supports the tool of motor control development support 'Renesas Motor Workbench'. With the tool, you can confirm internal data of software and use as user interface. Please utilize to choose the MCU and develop software with reference of this sample program in setting of peripherals or measurement of period of interrupt process.

In this document, the two motors are called Motor 1 and Motor 2.

2. Development environment

2.1 Test environments

Table 2-1 and Table 2-2 show the development environment of the software explained in this application note,

Table 2-1 Hardware development environment

Category	Product used
Microcontroller / CPU card product type	RA6T1 (R7FA6T1AD3CFP) / RTK0EMA170C00000BJ
Inverter board	48V inverter board / RTK0EM0000B10020BJ (2set)
Motor	TG-55L-KA (2set)

Table 2-2 Software development environment

e2studio version	FSP version	Toolchain version
V2023-07	V4.6.0	GCC ARM Embedded: V10.3.1.20210824

For purchase and technical support, contact Sales representatives and dealers of Renesas Electronics Corporation.

2.2 Hardware specifications

Overview of this system is explained below.

This APN describes two type hardware configurations.

1. Not using the extender board:
   Several Board modifications needed, and an own's cable needs to be prepared.
2. Using the extender board:
   A few Boards modification needed, and no own's cable needs to be prepared as commercially available cable can be used.

2.2.1 Hardware configuration diagram

(1) Overall configuration
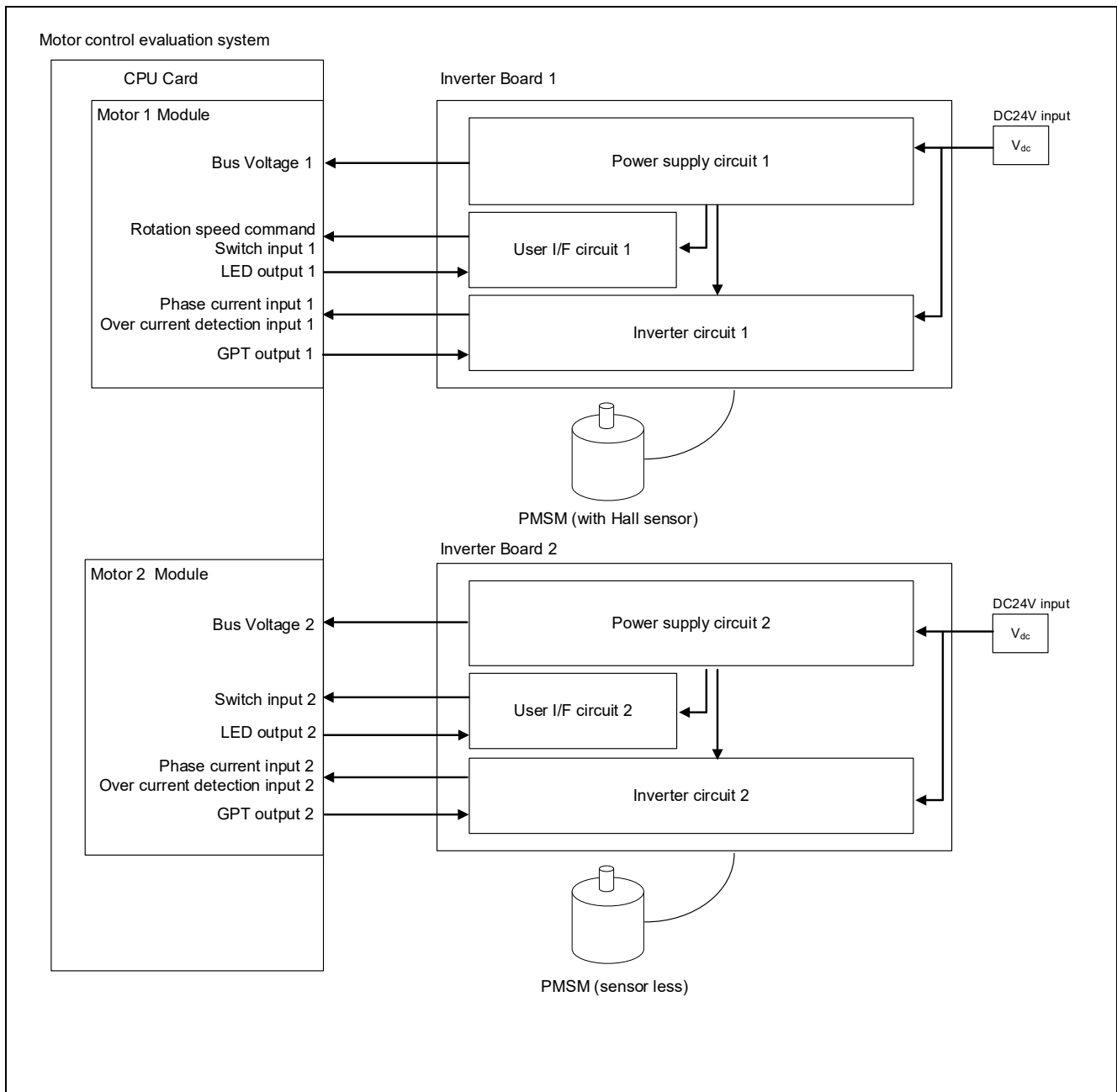
Figure 2-1 Hardware configuration diagram

(2)   Motor 1 module configuration
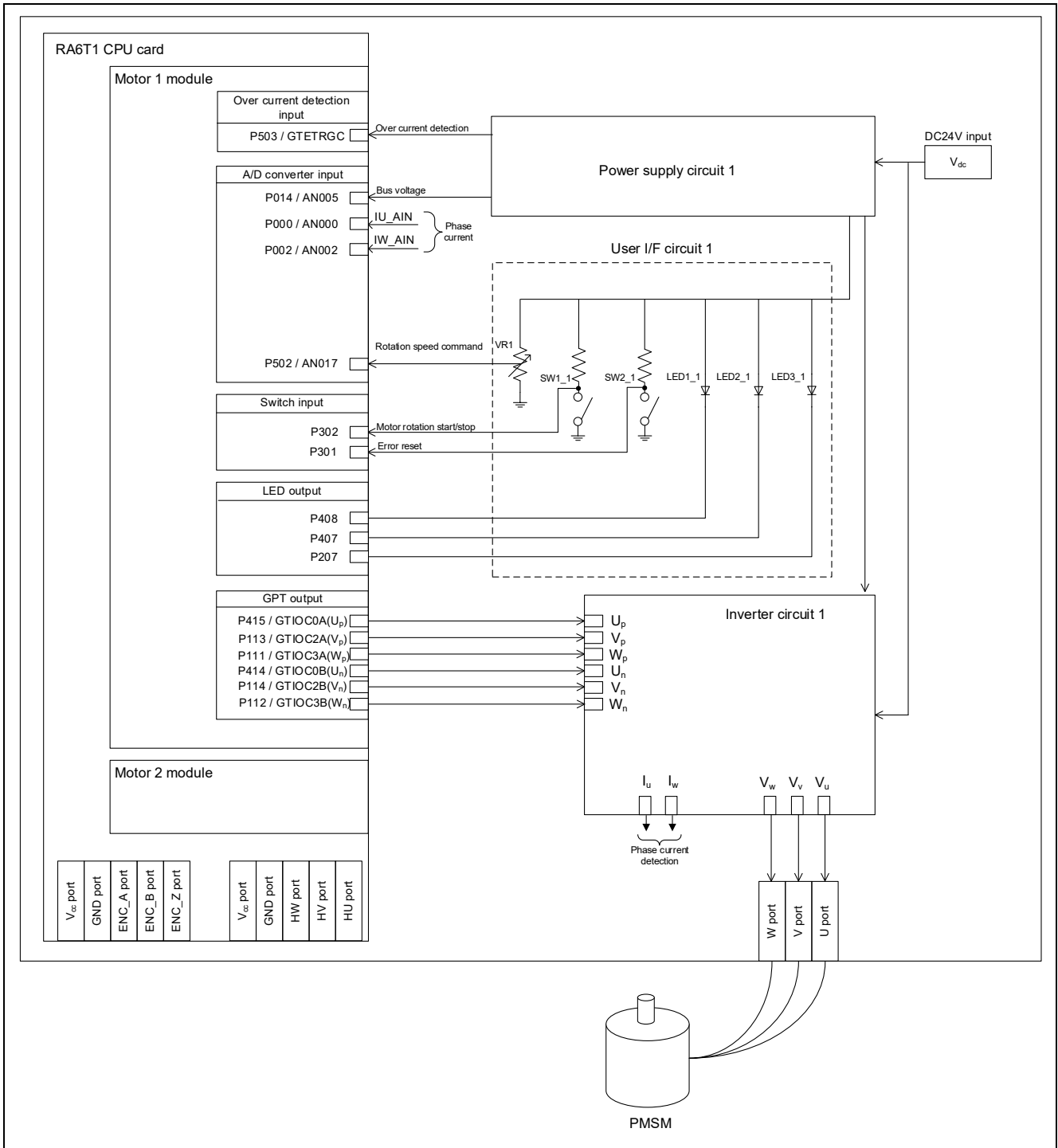
Figure 2-2 Hardware configuration diagram (Motor 1 module)

(3)　Motor 2 module configuration
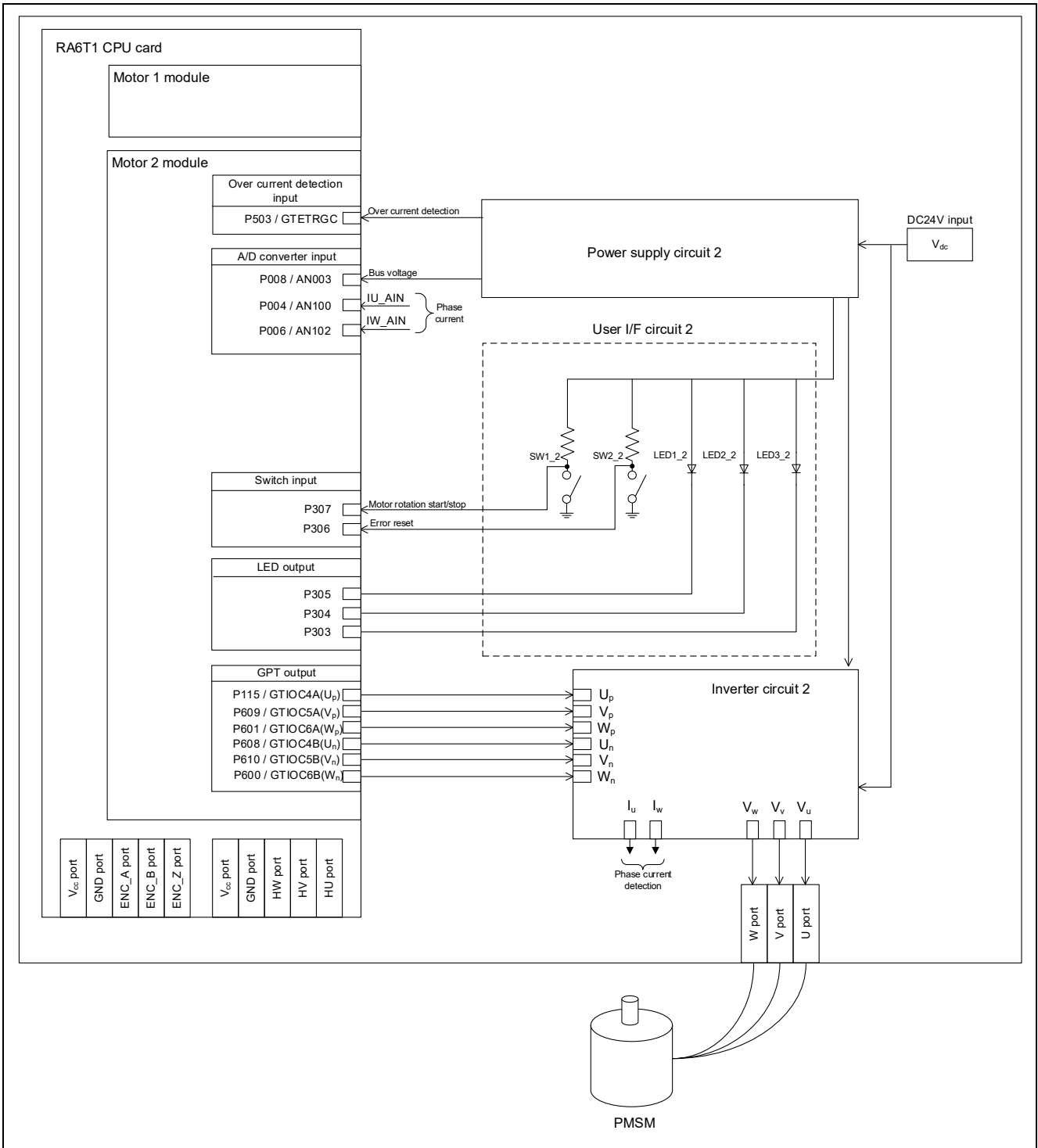
Figure 2-3 Hardware configuration diagram (Motor 2 module)

2.2.2 Inverter board connection

A hardware modification is required to use this system. Here, we will introduce you to each modification method mentioned above (1. Not using an extender board, 2. Using an extender board). The parts introduced in this chapter are just examples. Be sure to check the parts to be used by yourself.

2.2.2.1 Not using the extender board

(1) Mount the pin header on the "CNE" of the inverter board.

(ex.)Molex 901310945
2.54mm Pitch C-Grid III
Modular Crimp Housing,
Dual Row, 50 Circuits, Black
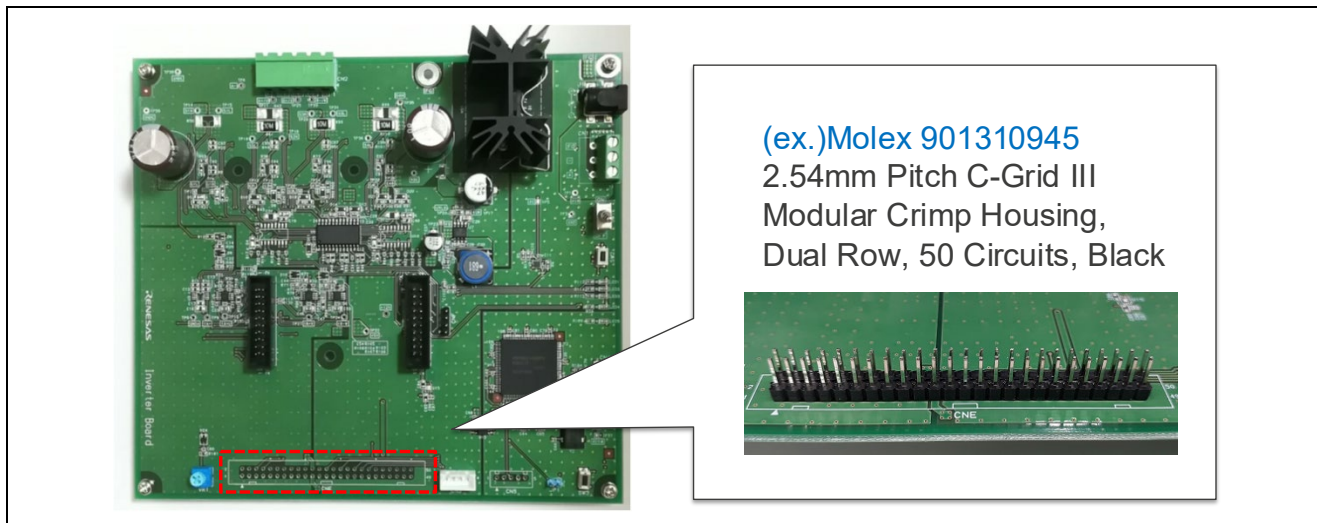
Figure 2-4 Mounting the pin header to the inverter board 2

(2) Mount the 0Ω resistor (size : 0603(mm)) at the R140, R142, R144, R145, R146, R147, R148, R149, R150, R152, R153, R154, R155, R156 on the inverter board 2.
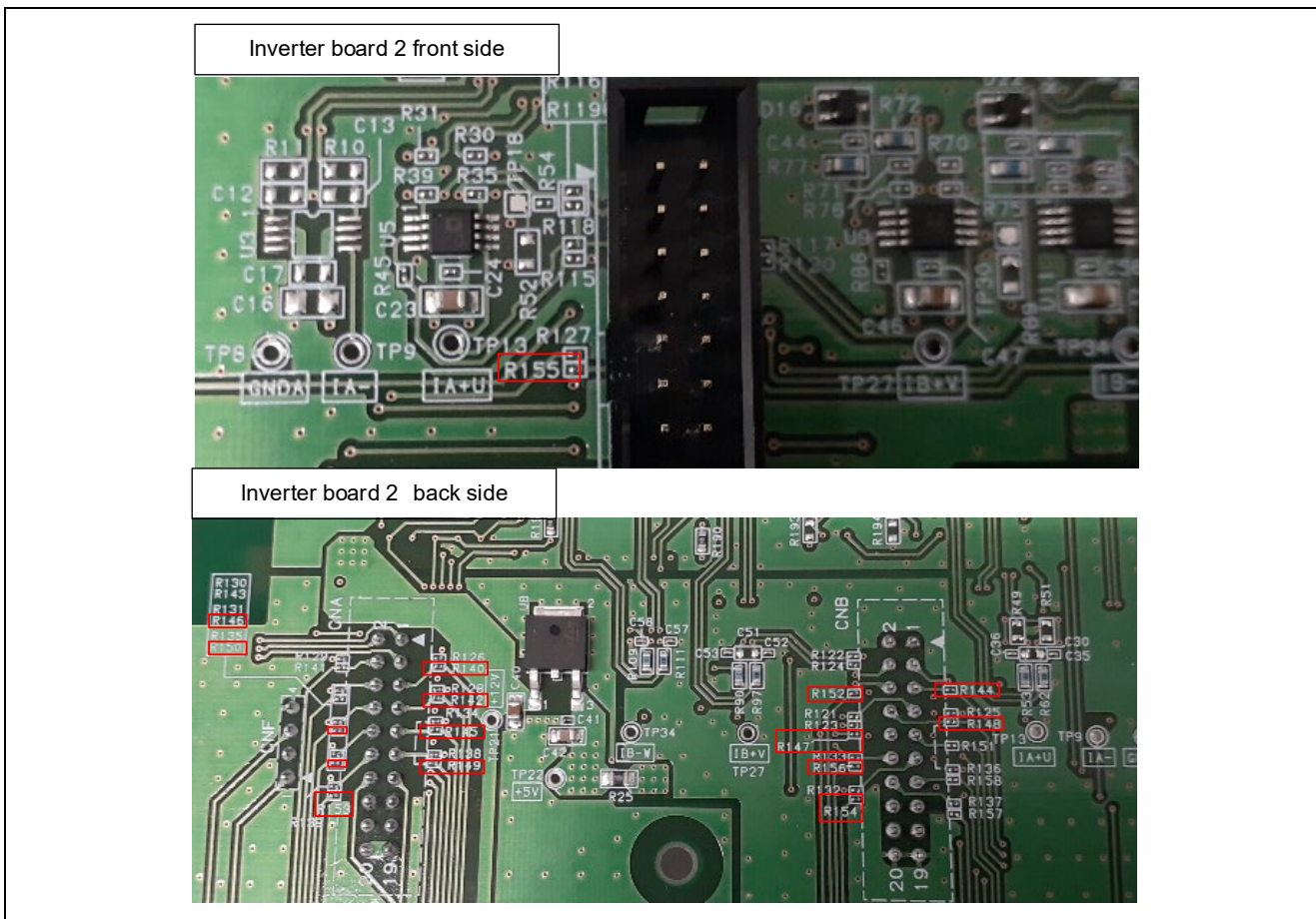
Figure 2-5 location of resistors which need to be mounted on inverter board 2

(3) Mount the pin header on the "CNC" of the CPU card.
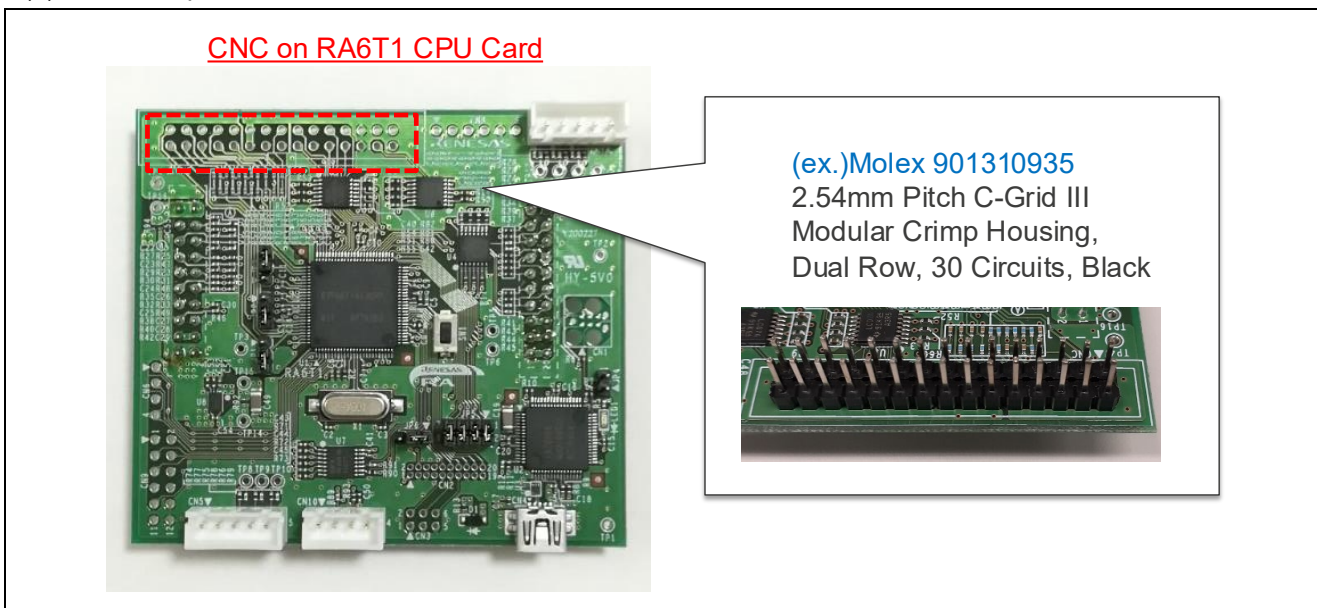
Figure 2-6 Mounting the pin header to the CPU card

(4) Connect the corresponding signal between the inverter board and the CPU card.
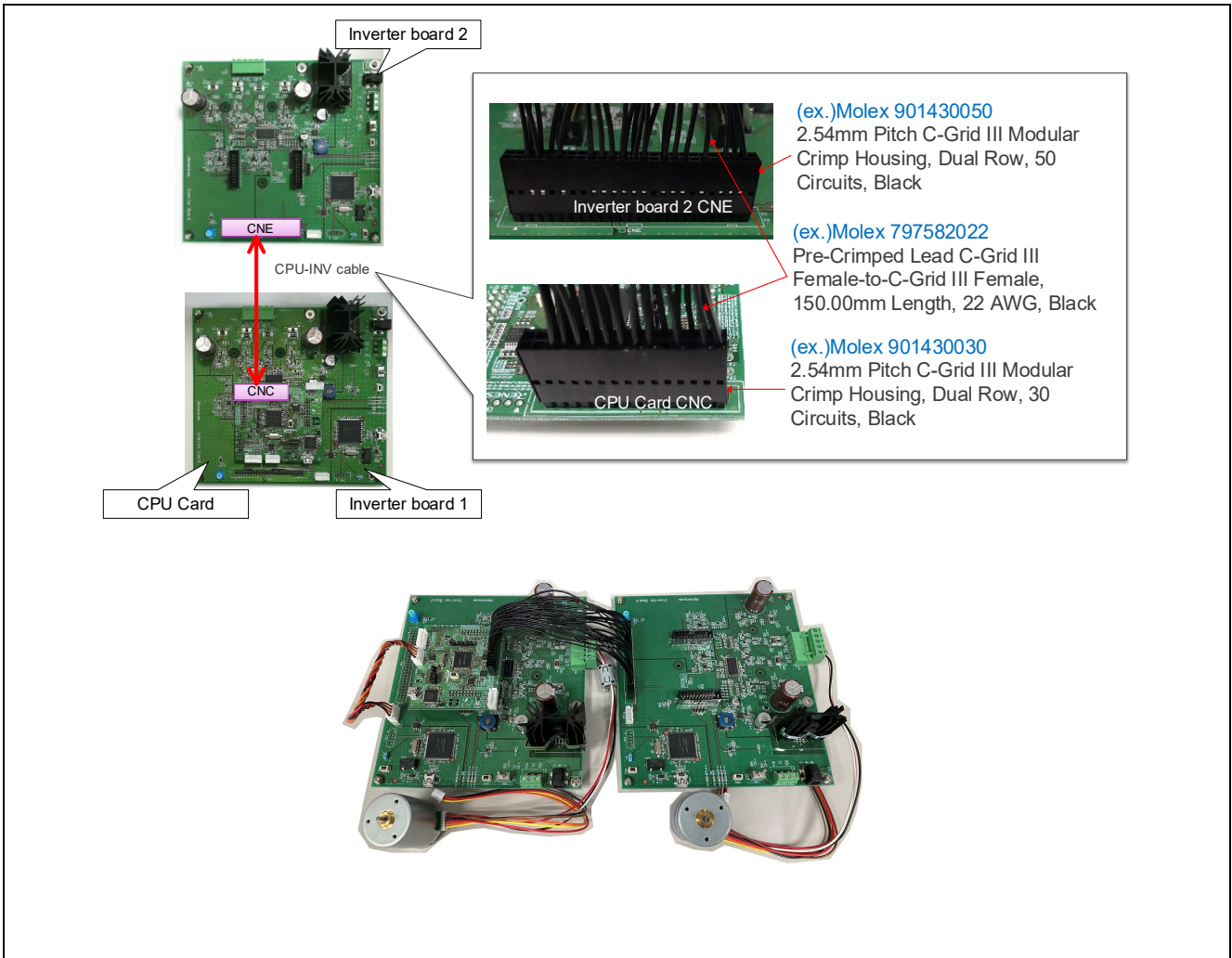
Figure 2-7 Connection between the inverter board and the CPU card

Table 2-3 Pin assignment correspondence table

INV (CNE)		RA6T1(CNC)	
Pin#	Pin name	Pin#	Pin name
3	GND_A	1	PGAVSS
12	VDC_EX	2	VPN
16	I_U/A+_EX	3	IU
18	I_V/B+_EX	4	IV
20	I_W/B-_EX	5	IW
4	V_U/A+_EX	7	VU
6	V_V/B+_EX	8	VV
8	V_W/B-_EX	9	VW
23	GND_D	11	VSS
19	GND_A	12	VSS
41	LED1	13	LED1
44	LED2	14	LED2
43	LED3	15	LED3
40	FO#_EX	16	OC#
24	PWM_WL/B-L_EX	17	WN
26	PWM_VL/B+L_EX	18	VN
28	PWM_UL/A+L_EX	19	UN
30	PWM_WH/B-_EX	20	WP
32	PWM_VH/B+H_EX	21	VP
34	PWM_UH/A+H_EX	22	UP
46	SW1	23	SW1
45	SW2	24	SW2
25	GND_D	25	VSS
27	GND_D	26	VSS

2.2.2.2 Using an extender board

If you use the extender board that fits the connector accommodation, there are only a few modification in the board and it can be realized with commercially available cable with one-to-one connection. For the extender board, refer to the information (CAD data, circuit diagram, BOM List) included in the APN for development.

(1) Mount the pin header on the "CNC" of the CPU card.

CNC on RA6T1 CPU Card

(ex.)Molex 901310935
2.54mm Pitch C-Grid III
Modular Crimp Housing,
Dual Row, 30 Circuits, Black
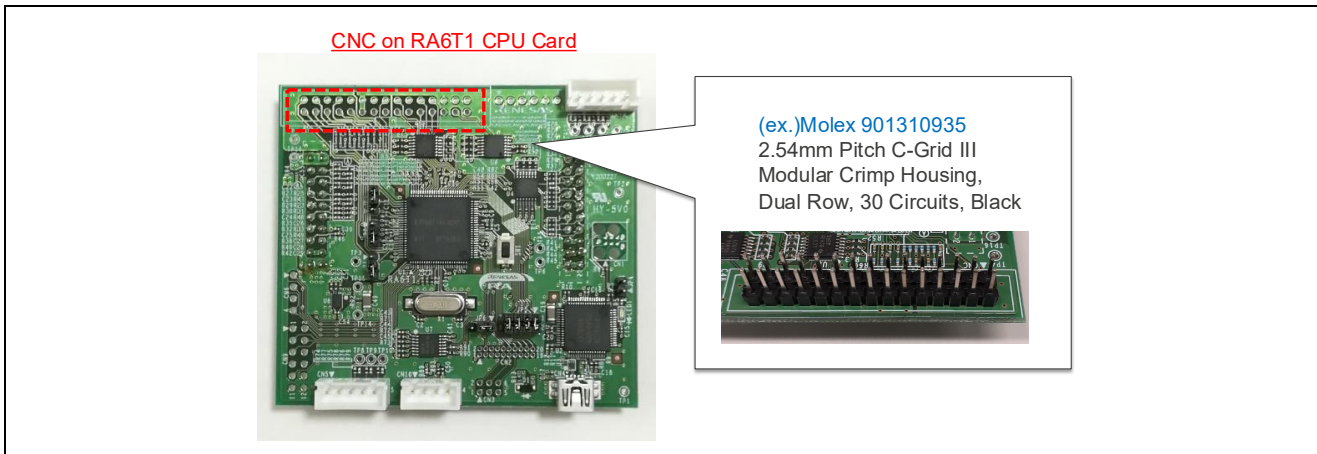
Figure 2-8 Mounting a pin header to the CPU card

(2) Connect the corresponding signal between the extender board and the CPU card.
Since the CNC of the extender board is 34pin, mount the unshrouded connector to connect with the CNC of 30pin RA6T1. When connecting with the cable, make sure that the PIN 1 of each board is aligned.

Extender board

Inverter board 2

CNC

1PIN

CPU-INV cable

CNC

1PIN

Inverter board 1

CPU card

Image of complete system
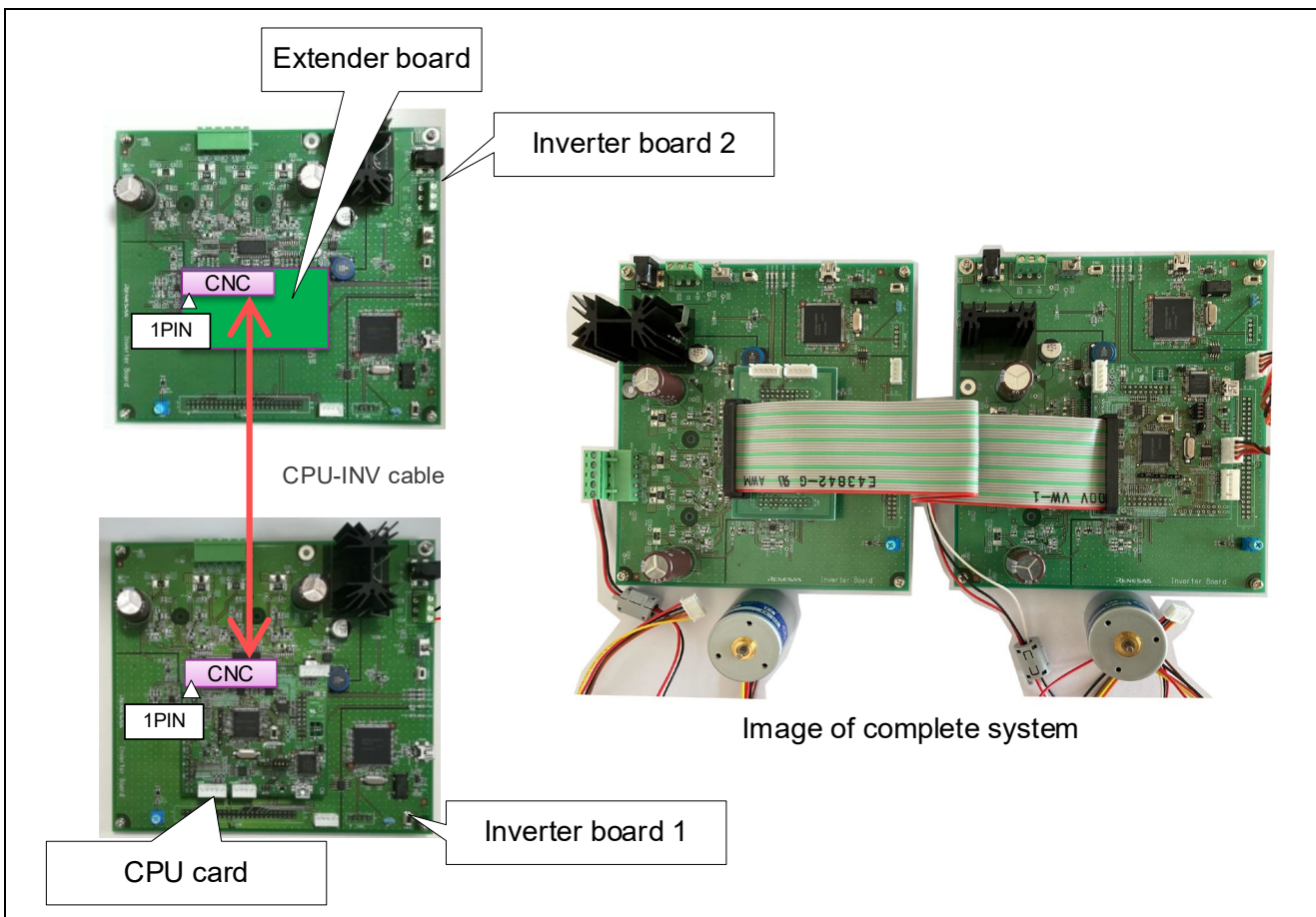
Figure 2-9 Connection between the extender board and the CPU card

Table 2-4 Pin assignment correspondence table

Extender board (CNC)		RA6T1(CNC)	
Pin#	Pin name	Pin#	Pin name
1	PGAVSS_2	1	PGAVSS
2	VPN_2	2	VPN
3	IU_2	3	IU
4	IV_2	4	IV
5	IW_2	5	IW
6	TEMP_2	6	NC
7	VU_2	7	VU
8	VV_2	8	VV
9	VW_2	9	VW
10	VR_2	10	NC
11	GND	11	VSS
12	GND	12	VSS
13	LED1#_2	13	LED1
14	LED2#_2	14	LED2
15	LED3#_2	15	LED3
16	FO#_2	16	OC#
17	WN_2	17	WN
18	VN_2	18	VN
19	UN_2	19	UN
20	WP_2	20	WP
21	VP_2	21	VP
22	UP_2	22	UP
23	SW1#_2	23	SW1
24	SW2#_2	24	SW2
25	GND	25	VSS
26	GND	26	VSS
27	ENC_A_2	27	P602
28	ENC_B_2	28	NC
29	ENC_Z_2	29	NC
30	GND	30	NC
31	GND	-	-
32	HALL_U_2	-	-
33	HALL_V_2	-	-
34	HALL_W_2	-	-

2.2.3 User interface

Table 2-5 is the list of user interface of this system.

Table 2-5 User interfaces

Item	Interface component	Function
Rotation speed command	Variable resistor of Motor 1 side (VR1_1)	Reference value of rotation speed input (analog value)
START / STOP 1	Toggle switch of Motor 1 side (SW1_1)	Motor 1 rotation start/stop command
ERROR RESET 1	Push switch of Motor 1 side (SW2_1)	Command of recovery from error status
LED1_1	Orange LED of Motor 1 side (LED1_1)	• At the time of Motor 1 rotation: ON • At the time of Motor 1 stop: OFF
LED2_1	Orange LED of Motor 1 side (LED2_1)	• At the time of error detection: ON • At the time of normal operation: OFF
START / STOP 2	Toggle switch of Motor 2 side (SW1_2)	Motor 2 rotation start/stop command
ERROR RESET 2	Push switch of Motor 2 side (SW2_2)	Command of recovery from error status
LED1_2	Orange LED of Motor 2 side (LED1_2)	• At the time of Motor 2 rotation: ON • At the time of Motor 2 stop: OFF
LED2_2	Orange LED of Motor 2 side (LED2_2)	• At the time of error detection: ON • At the time of normal operation: OFF
RESET	Push switch (RESET1)	System reset

Table 2-6 and Table 2-7 are the lists of port interface of this system.

Table 2-6 Port interfaces (Motor 1 side)

Function	RA6T1
Inverter bus voltage measurement	P014 / AN005
For rotation speed command value input (VR1_1)	P502 / AN017
START/STOP toggle switch (SW1_1)	P302
ERROR RESET push switch (SW2_1)	P301
LED1_1 ON/OFF control	P408
LED2_1 ON/OFF control	P407
U1 phase current measurement	P000 / AN000
W1 phase current measurement	P002 / AN002
PWM output (Up1)	P415 / GTIOC0A
PWM output (Vp1)	P113 / GTIOC2A
PWM output (Wp1)	P111 / GTIOC3A
PWM output (Un1)	P414 / GTIOC0B
PWM output (Vn1)	P114 / GTIOC2B
PWM output (Wn1)	P112 / GTIOC3B
PWM emergency stop input at the time of overcurrent detection	P503 / GTETRGC

Table 2-7 Port interfaces (Motor 2 side)

Function	RA6T1
Inverter bus voltage measurement	P008 / AN003
START/STOP toggle switch (SW1_2)	P307
ERROR RESET push switch (SW2_2)	P306
LED1_2 ON/OFF control	P305
LED2_2 ON/OFF control	P304
U2 phase current measurement	P004 / AN100
W2 phase current measurement	P006 / AN102
PWM output (Up2)	P115 / GTIOC4A
PWM output (Vp2)	P609 / GTIOC5A
PWM output (Wp2)	P601 / GTIOC6A
PWM output (Un2)	P608 / GTIOC4B
PWM output (Vn2)	P610 / GTIOC5B
PWM output (Wn2)	P600 / GTIOC6B
PWM emergency stop input at the time of overcurrent detection	P104 / GTETRGB

2.2.4   Peripheral functions

List of the peripheral functions used in this system is given in Table 2-8.

Table 2-8 List of the peripheral functions

Peripheral	Purpose	RA6T1
A/D converter	U phase current measurement_1	AN000
	V phase current measurement_1	AN001
	W phase current measurement_1	AN002
	U phase current measurement_2	AN100
	V phase current measurement_2	AN101
	W phase current measurement_2	AN102
	Inverter bus voltage measuremen_1	AN005
	Inverter bus voltage measurement_2	AN003
	For rotation speed command value input (analog value)	AN017
AGT	Speed control interval timer_1	AGT0
	Speed control interval timer_2	AGT1
GPT	U phase PWM output_1	CH0
	V phase PWM output_1	CH2
	W phase PWM output_1	CH3
	U phase PWM output_2	CH4
	V phase PWM output_2	CH5
	W phase PWM output_2	CH6
POEG	PWM emergency stop input at the time of overcurrent detection_1	Group C
	PWM emergency stop input at the time of overcurrent detection_2	Group B

(1)   A/D Converter (ADC)

U-phase current, W-phase current, inverter bus voltage, and rotation speed command for Motor 1 and 2 are measured in "Single Scan Mode" (use a hardware trigger). A/D conversion is implemented to be synchronized with carrier synchronized interrupt.

(2)   Low Power Asynchronous General-Purpose Timer (AGT)

The AGT is used as 1 [msec] interval timer.

(3)   General PWM Timer (GPT)

On the channel 0,2,3,4,5 and 6, output with dead time is performed by using the complementary PWM Output Operating Mode.

(4)   Port Output Enable for GPT (POEG)

The port executing PWM output are set to high impedance state when an overcurrent is detected (when a low level of the GTETRGC/GTETRGD port is detected).

Figure 2-10 Overall FSP stacks diagram
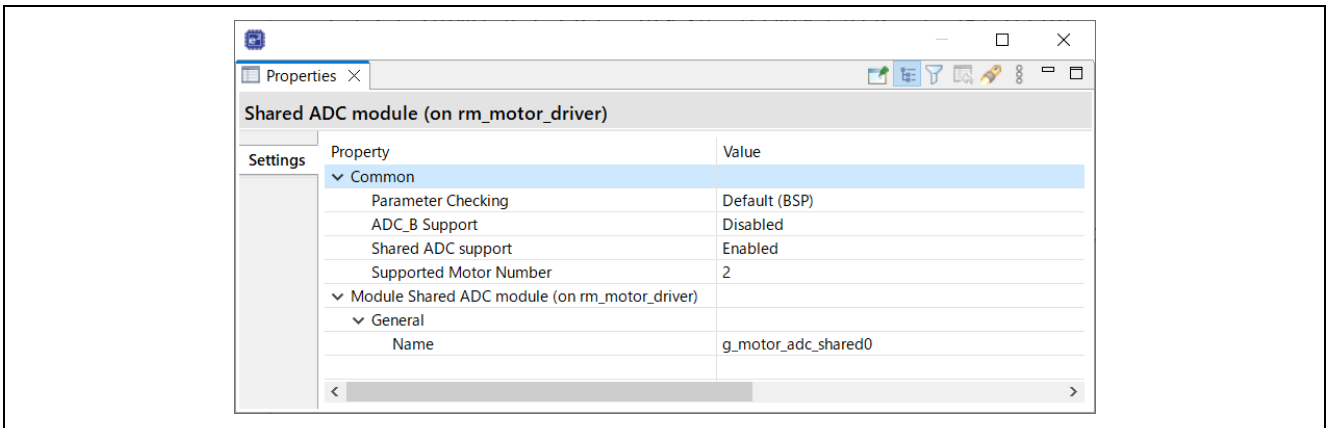
Figure 2-11 FSP configuration of ADC shared module
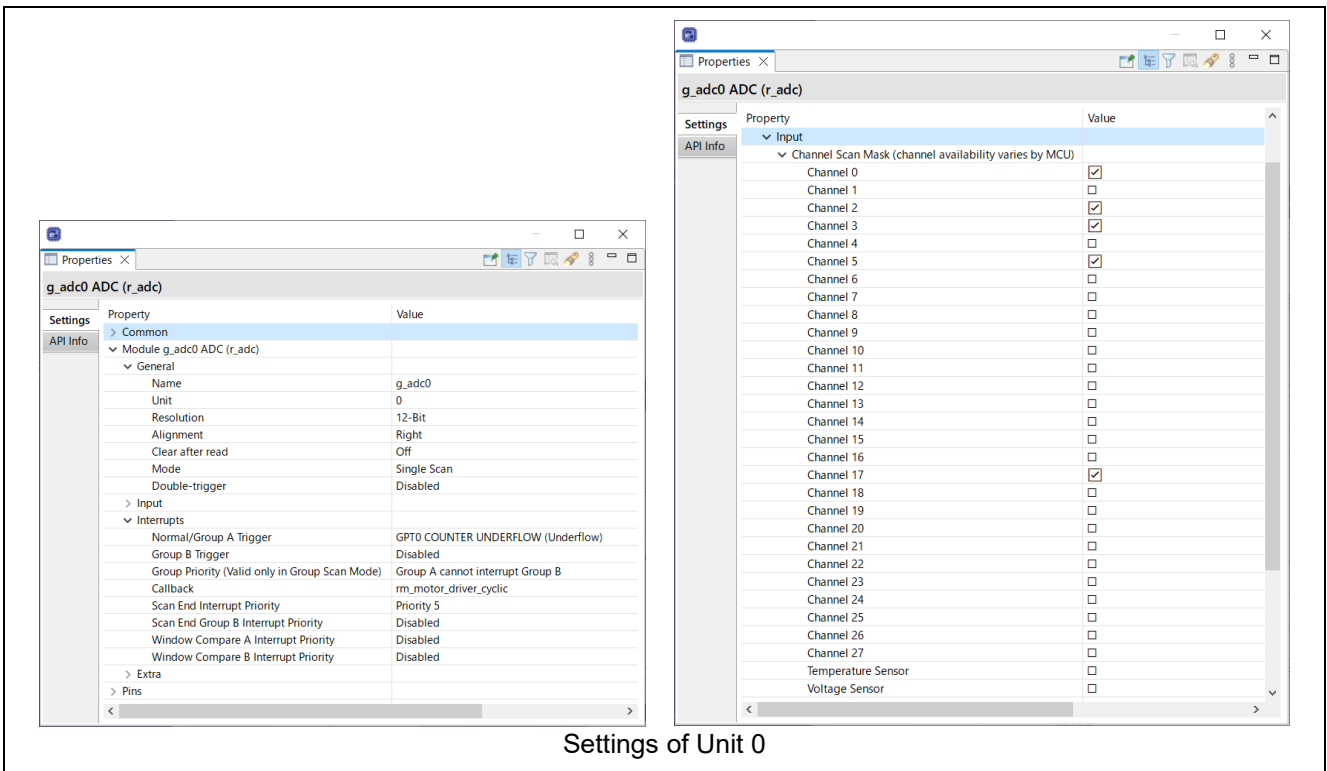
Figure 2-12 FSP configuration of ADC driver [1/2]

Settings of Unit 1
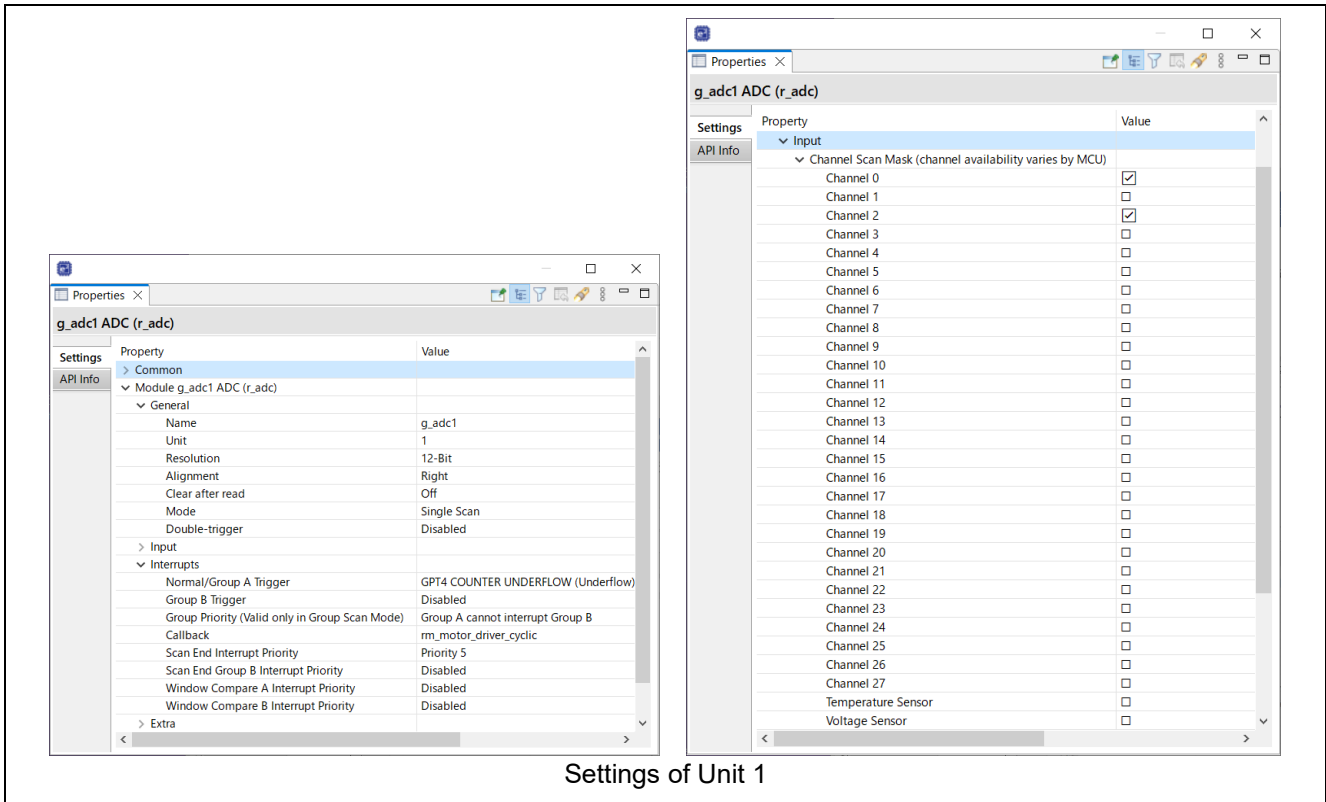
Figure 2-13 FSP configuration of ADC driver [2/2]

Motor 1   Motor 2
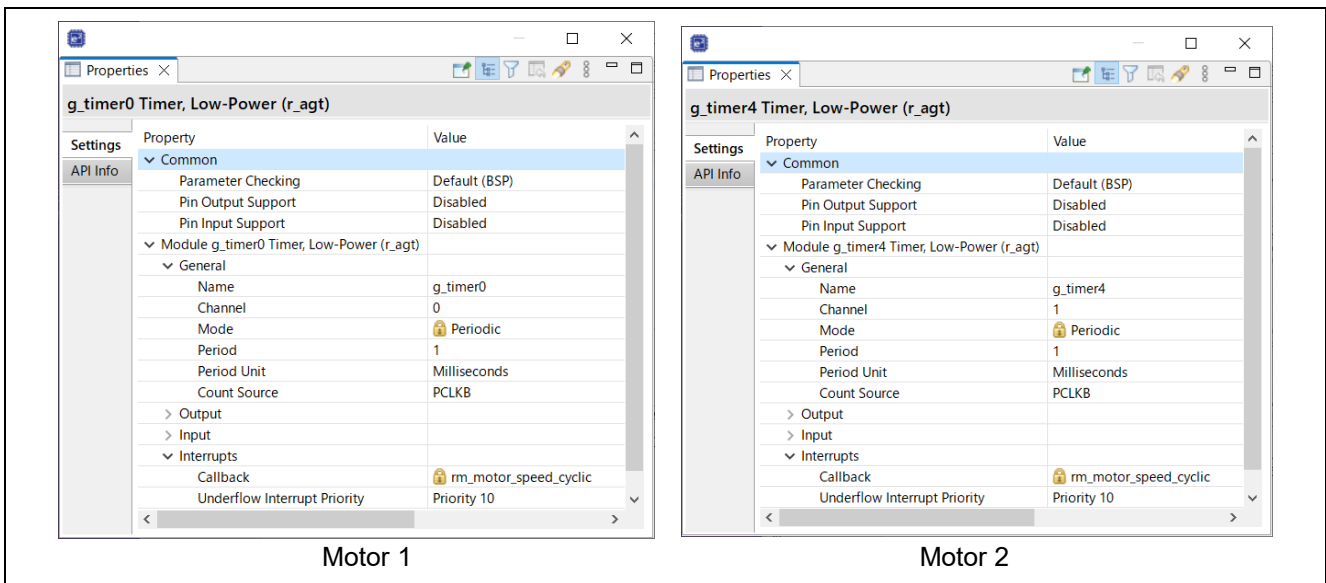
Figure 2-14 FSP configuration of AGT driver

Motor 1 U phase setting (example)

Figure 2-15 FSP configuration of GPT driver

Motor 1	Motor 2

Figure 2-16 FSP Configuration of POEG driver

2.3   Software configuration

2.3.1   Software file configuration

Folder and file configuration of the software is given below.

Table 2-9 File and folder configuration [1/2]

Folder	Subfolder	File	Remarks
ra_cfg			Generated config header
ra_gen			Generated register setting, main function etc
ra	arm		CMSIS source code
	board		Function definition for board
	fsp/inc/api	bsp_api.h	BSP API definition
		r_adc_api.h	AD API definition
		r_ioport_api.h	I/O API definition
		r_poeg_api.h	POEG API definition
		r_three_phase_api.h	3phase PWM API definition
		r_timer_api.h	Timer API definition
		r_transfer_api.h	Transfer API definition
		rm_motor_angle_api.h	Angle API definition
		rm_motor_api.h	Motor API definition
		rm_motor_current_api.h	Current API definition
		rm_motor_driver_api.h	Motor driver API definition
		rm_motor_speed_api.h	Speed API definition
	fsp/inc/instances	r_adc.h	Function definition for AD
		r_agt.h	Function definition for AGT
		r_gpt_three_phase.h	Function definition for 3 Phase PWM
		r_gpt.h	Function definition for GPT
		r_ioport.h	Function definition for I/O
		r_poeg.h	Function definition for POEG
		rm_motor_current.h	Function definition for current control
		rm_motor_driver.h	Function definition for motor driver
		rm_motor_estimate.h	Function definition for angle estimate
		rm_motor_sensorless.h	Function definition for Sensorless
		rm_motor_speed.h	Function definition for Speed

Table 2-10 File and folder configuration [2/2]

Folder	Subfolder	File	Remarks
ra	fsp/lib		Library files
	fsp/src	bsp	BSP driver
		r_adc/r_adc.c	AD driver
		r_agt/r_agt.c	AGT driver
		r_gpt/r_gpt.c	GPT driver
		r_gpt_three_phase/r_gpt_three_phase.c	3 phase PWM driver
		r_ioport/r_ioport.c	I/O driver
		r_poeg/r_poeg.c	POEG driver
		rm_motor_current/rm_motor_current.c	Current control driver
		rm_motor_current/rm_motor_current_library.h	Current control library API definition
		rm_motor_estimate/rm_motor_estimate.c	Rotor angle/speed estimation driver
		rm_motor_estimate/rm_motor_estimate_library.h	Rotor angle/speed estimation library API definition
		rm_motor_sensorless/rm_motor_sensorless.c	Sensorless driver
		rm_motor_speed/rm_motor_speed.c	Speed control driver
		rm_motor_speed/rm_motor_speed_library.h	Speed control library API definition
src	application/main	mtr_main.h , mtr_main.c	User main function
		r_mtr_control_parameter.h	Control parameters definition
		r_mtr_motor_parameter.h	Motor parameters definition
	application/ user_interface/ics	r_mtr_ics.h , r_mtr_ics.c	Function definition for Analyzer
		ICS2_RA6T1.h	Function definition for GUI tool
		ICS2_RA6T1.o	Communication library for GUI tool

RENESAS

2.3.2 Module configuration

Module configuration of the software is described below.
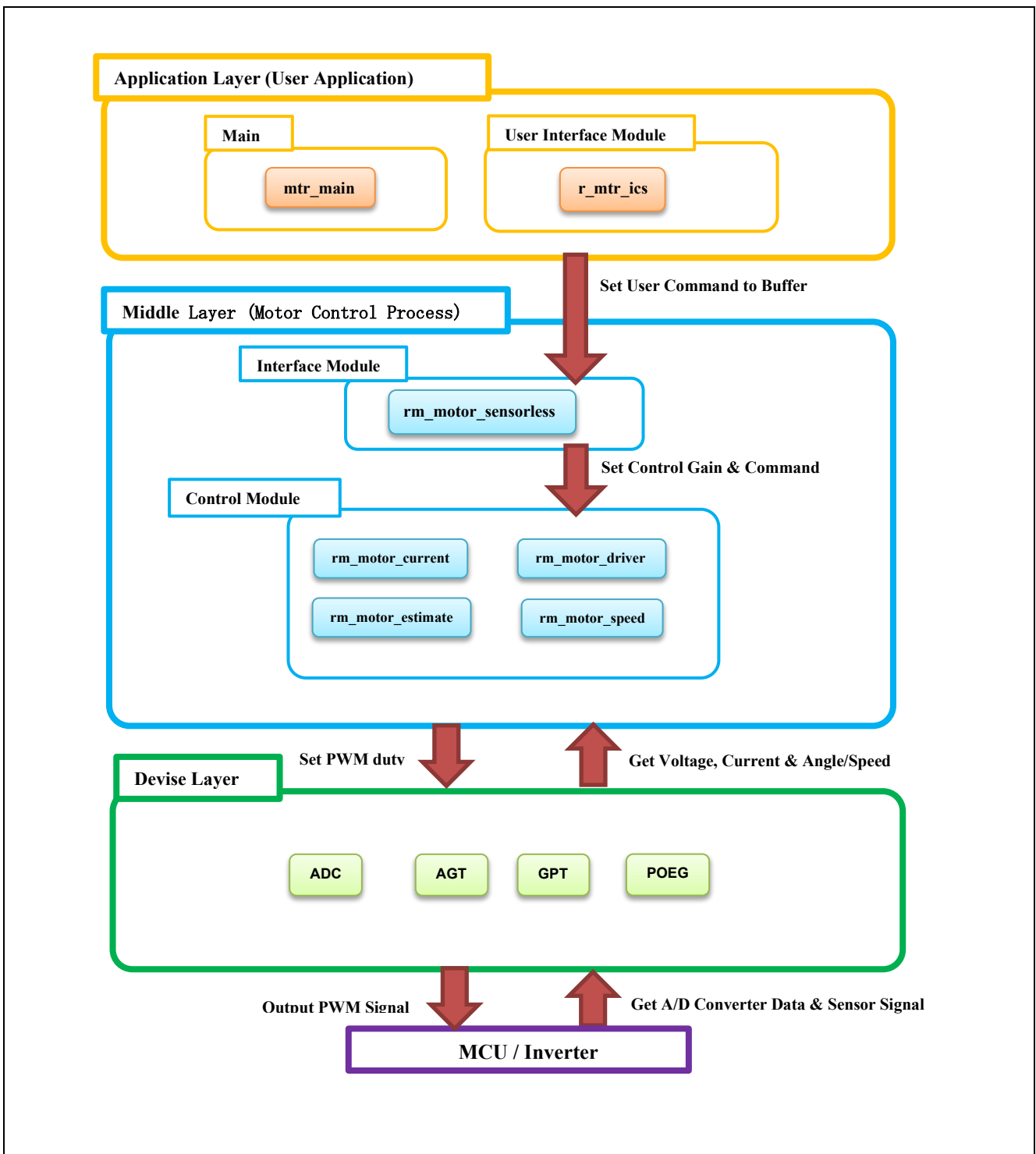
Figure 2-17 Module configuration

2.4 Software Specifications

Table 2-11 shows basic software specification of this system.

Specifications are the same for Motor 1 and 2, except for hardware interface.

Table 2-11 Basic specifications of sensorless vector control software

Item	Content	
Control method	Vector control	
Position detection method	Sensorless	
Motor rotation start/stop	Determined depending on the level of SW1_1 for Motor 1 and SW1_2 for Motor 2, or input from Renesas Motor Workbench	
Input voltage	DC 24V	
Main clock frequency	120 [MHz]	
Carrier frequency (PWM)	20 [kHz] (Carrier period: 50 [μs])	
Dead time	2 [μs]	
Control period	Current control / Position and speed estimation: 100 [μs] Speed control: 1 [ms]	
Rotation speed control range	CW : 0 [rpm] to 2650 [rpm] CCW : 0 [rpm] to 2650 [rpm] However, 600 [rpm] or less is driven by a speed open loop.	
Natural frequency of each control system	Current control system : 300 [Hz] Speed control system : 5 [Hz] BEMF estimation system : 1000 [Hz] Position estimation system : 50 [Hz]	
Optimization setting of compiler	Optimization level	Optimize more(-O2) (default setting)
Processing stop for protection	[Motor 1] Disables the Motor 1 control signal output (six outputs), under any of the following conditions. 1. Instantaneous value of current of any phase exceeds 0.89(=0.42*sqrt (2)*1.5) [A] (monitored in current control period) 2. Inverter bus voltage exceeds 28 [V] (monitored in current control period) 3. Inverter bus voltage is less than 14 [V] (monitored in current control period) 4. Rotation speed exceeds 3000 [rpm] (monitored in current control period) When an external over current signal is detected (when a low level is detected), the PWM output ports are set to high impedance state.	[Motor 2] Disables the Motor 2 control signal output (six outputs), under any of the following conditions. 1. Instantaneous value of current of any phase exceeds 0.89(=0.42*sqrt (2)*1.5) [A] (monitored in current control period) 2. Inverter bus voltage exceeds 28 [V] (monitored in current control period) 3. Inverter bus voltage is less than 14 [V] (monitored in current control period) 4. Rotation speed exceeds 3000 [rpm] (monitored in current control period) When an external over current signal is detected (when a low level is detected), the PWM output ports are set to high impedance state.

2.5 Interrupt Priority

Table 2-12 shows the interrupt and priorities used in this system.

Table 2-12 Interrupt priority

Interrupt level	Priority	Function
15	Min	
14		
13		
12		
11		
10		AGT0/1 INT 1 [msec] Interrupt handling (Motor 1, 2 are same priority)
9		
8		
7		
6		
5		ADC0 ADI0/2 A/D complete interrupt (Motor 1, 2 are same priority)
4		
3		
2		
1		
0	Max	POEG1/2 EVENT Over current error interrupt (Motor 1, 2 are same priority)

Allocations

Interrupt	Event	ISR
0	AGT0 INT (AGT interrupt)	agt_int_isr
1	ADC0 SCAN END (A/D scan end interrupt)	adc_scan_end_isr
2	ADC1 SCAN END (A/D scan end interrupt)	adc_scan_end_isr
3	POEG2 EVENT (Port Output disable interrupt C)	poeg_event_isr
4	AGT1 INT (AGT interrupt)	agt_int_isr
5	POEG1 EVENT (Port Output disable interrupt B)	poeg_event_isr

Figure 2-18 FSP Interrupts configuration

3. Descriptions of Control Program

3.1 Contents of Control

3.1.1 Motor start/stop

The start and stop of the motor are controlled by input from Renesas Motor Workbench or SW1_1, SW1_2.

SW1_1 and SW1_2 are assigned to a general-purpose port. When the port is at a "High" level, it is determined that the start switch is being pressed. Conversely, when the level is switched to "Low", the software determines that the motor should be stopped.

"High" level: Motor Start

"Low" level: Motor Stop

3.1.2 A/D converter

(1) Motor rotation speed reference

The motor rotation speed reference can be set by Renesas Motor Workbench input or A/D conversion of the VR1 output value (analog value). The A/D converted value is used as rotation speed command value, as shown below.

Table 3-1 Conversion ratio of rotation speed reference

Item	Conversion ratio (Reference : A/D conversion value)	
Rotation speed reference	CW	0 [rpm] to 2650[rpm] : 0800H to 0FFFH
	CCW	0 [rpm] to 2650[rpm] : 07FFH to 0000H

(2) Inverter bus voltage

Inverter bus voltage is measured as given in Table 3-2.

It is used for modulation factor calculation and over-/low-voltage detection. (When an abnormality is detected, PWM is stopped.)

Table 3-2 Inverter bus voltage conversion ratio

Item	Conversion ratio (Inverter bus voltage : A/D conversion value)
Inverter bus voltage	0 [V] to 111 [V] : 0000H to 0FFFH

(3) U, W phase current

The U and W phase currents are measured as shown in Table 3-3 and used for vector control.

Table 3-3 Conversion ratio of U and W phase current

Item	Conversion ratio (U, W phase curren t: A/D conversion value)
U, W phase current	-12.5 [A] to 12.5 [A] : 0000H to 0FFFH (Note) Current = (5V-2.5V)/(0.01Ohm*20) =12.5A In this system, the current detection circuit shifts the level from 5V to 3V, so 0E8BH is the upper limit of A / D conversion.

3.1.3 Modulation (current control module)

A modulated voltage can be output to improve the efficiency of voltage usage. The modulation operation is set from the API of the current control module.

3.1.3.1 Sine wave modulation (MOD_METHOD_SPWM)

The modulation factor m is defined as follows.

$$m = \frac{V}{E}$$

m：Modulation ratio   V：Reference voltage   E：Inverter input voltage

3.1.3.2 Space Vector Modulation (MOD_METHOD_SVPWM)

In vector control of a permanent magnet synchronous motor, generally, the desired voltage command value of each phase is generated sinusoidally. However, if the generated value is used as-is for the modulation wave for PWM generation, voltage utilization as applied to the motor (in terms of line voltage) is limited to a maximum of 86.7% with respect to inverter bus voltage. As such, as shown in the following expression, the average of the maximum and minimum values is calculated for the voltage command value of each phase, and the value obtained by subtracting the average from the voltage command value of each phase is used as the modulation wave. As a result, the maximum amplitude of the modulation wave is multiplied by $\sqrt{3}/2$, while voltage utilization becomes 100% and line voltage is unchanged.

$$\begin{pmatrix} V_u' \\ V_v' \\ V_w' \end{pmatrix} = \begin{pmatrix} V_u \\ V_v \\ V_w \end{pmatrix} + \Delta V \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

$$\because \Delta V = -\frac{V_{max} + V_{min}}{2} , V_{max} = max\{V_u, V_v, V_w\} , V_{min} = min\{V_u, V_v, V_w\}$$

V_u, V_v, V_w: Command values of U-, V-, and W-phases

V_u', V_v', V_w': Command values of U-, V-, and W-phases for PWM generation (modulation wave)

The modulation factor m is defined as follows.

$$m = \frac{V'}{E}$$

m：Modulation ratio   V'：Reference phase voltage for PWM
E：Inverter input voltage

3.1.4 State transition

Figure 3-1 is a state transition diagram of the sample software. In the target software of this application note, the software state is managed by "SYSTEM MODE". Motor 1 and 2 are controlled in the same method.
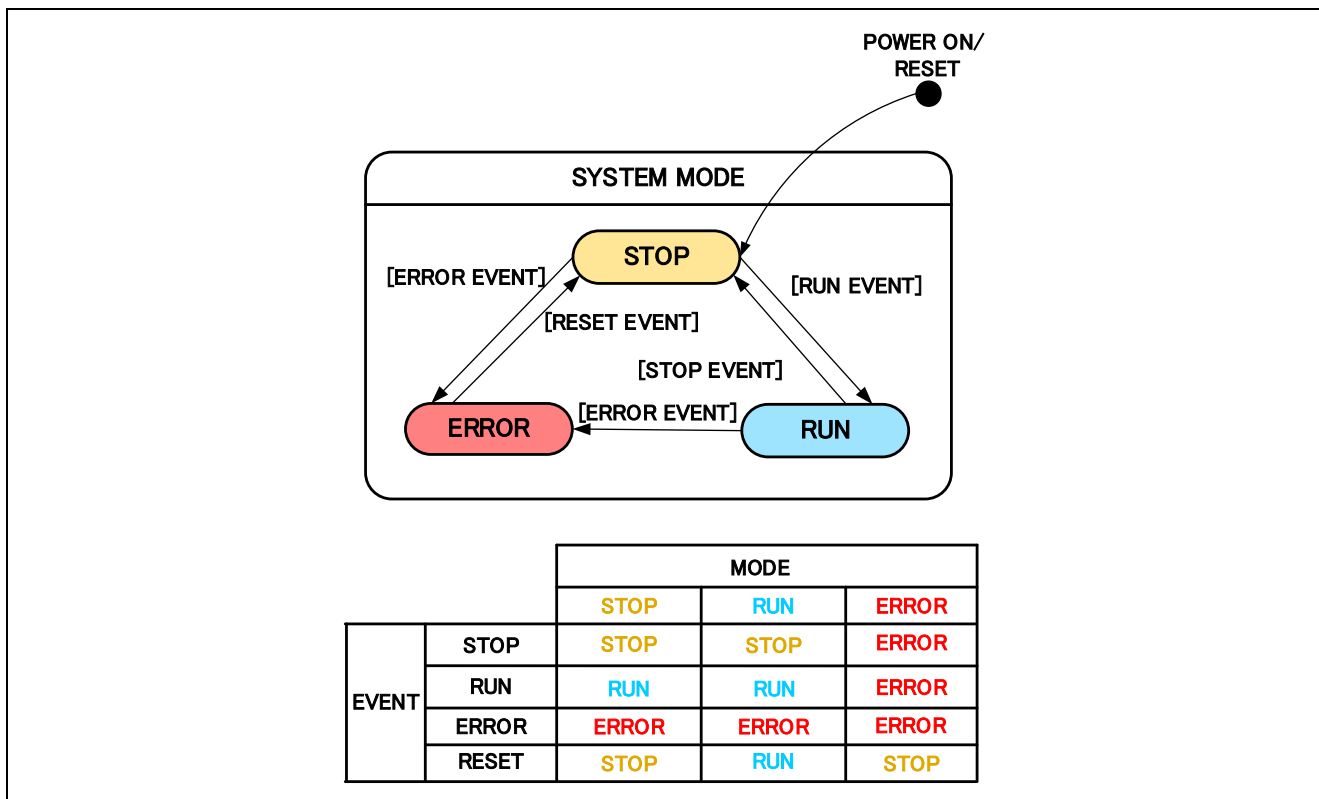
Figure 3-1 State transition diagram of sensorless vector control software

(1) SYSTEM MODE

"SYSTEM MODE" indicates the operating states of the system. The state transits on occurrence of each event (EVENT). "SYSTEM MODE" has 3 states that are motor drive stop (INACTIVE), motor drive (ACTIVE), and abnormal condition (ERROR).

(2) EVENT

When "EVENT" occurs in each "SYSTEM MODE", "SYSTEM MODE" changes as shown the table in Figure 3-1, according to that "EVENT". The occurrence factors of each event are shown below.

Table 3-4 List of EVENT

EVENT name	Occurrence factor
STOP	by user operation
RUN	by user operation
ERROR	when the system detects an error
RESET	by user operation

3.1.5　Start-up method

Figure 3-2 shows startup control of vector control software. Each mode is controlled by flags managing each reference of the d-axis current, q-axis current, and speed. Motor 1 and 2 are controlled in the same method.
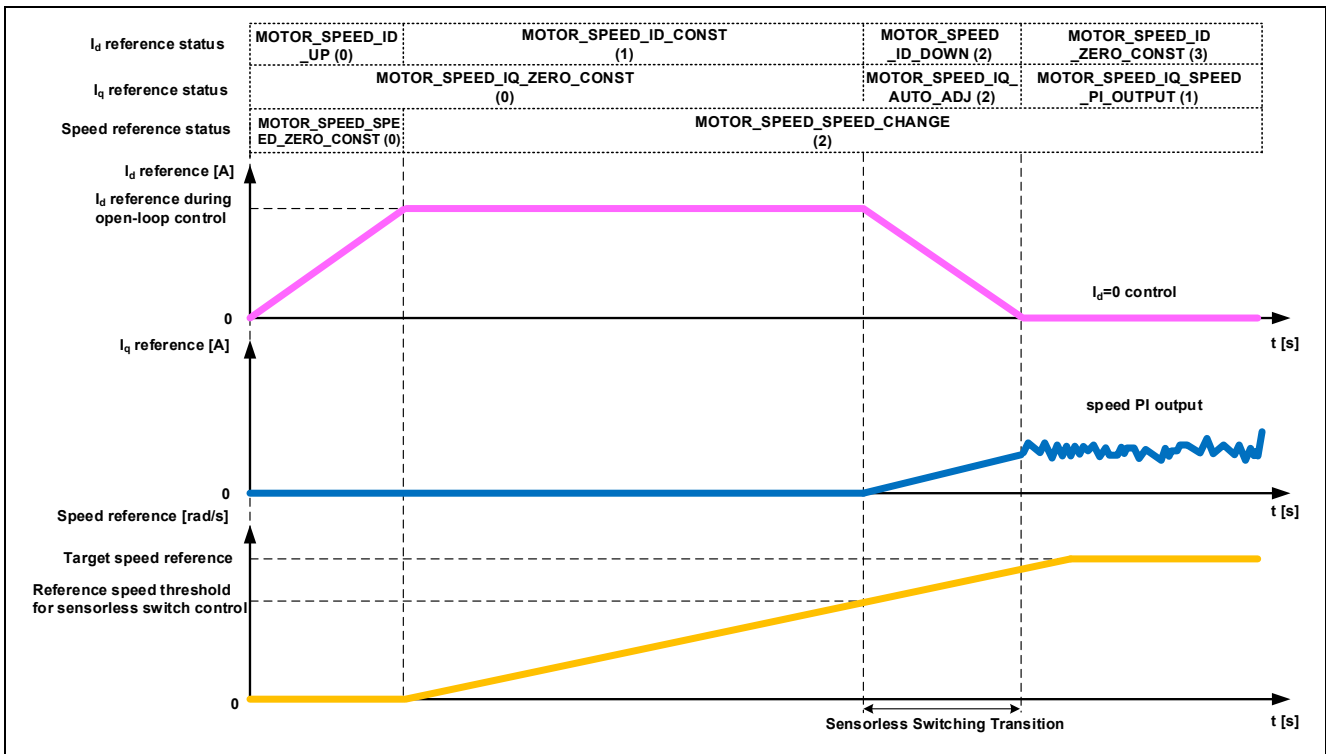
Figure 3-2 Startup control of vector control software

3.1.6 System protection function

This control software has the following error status and executes emergency stop functions in case of occurrence of respective errors. Table 3-5 shows each software threshold for the system protection function.

- Over current error

  The PWM output ports are set to high impedance state in response to an emergency stop signal (over current detection) from the hardware.

  In addition, U, V, and W phase currents are monitored in over current monitoring cycle. When an over current (when the current exceeds the over current limit) is detected, the CPU executes emergency stop (software detection).

- Over voltage error

  The inverter bus voltage is monitored in over voltage monitoring cycle. When an over voltage is detected (when the voltage exceeds the over voltage limit), the CPU performs emergency stop. Here, the over voltage limit is set in consideration of the error of resistance value of the detect circuit.

- Low voltage error

  The inverter bus voltage is monitored in low-voltage monitoring cycle. The CPU performs emergency stop when low voltage (when voltage falls below the limit) is detected. Here, the low voltage limit is set in consideration of the error of resistance value of the detect circuit.

- Over speed error

  The rotation speed is monitored in rotation speed monitoring cycle. The CPU performs emergency stop when the speed is over the limit.

Table 3-5 Setting values of the system protection function

Error name	Threshold		Monitoring cycle
Over current error	Over current limit [A]	0.89	Current control
Over voltage error	Over voltage limit [V]	28	Current control
Low voltage error	Low voltage limit [V]	14	Current control
Over speed error	Speed limit [rpm]	3000	Current control

3.1.7   Carrier synchronized interrupt

In the case of 2-motor control, if each timer is started at the same time, the interrupt timings will overlap, so the two motors should work in turn to make sure they enter the PWM interrupt at different time.

In this control program, it is implemented by adjusting the carrier cycle of Motor 2 side.
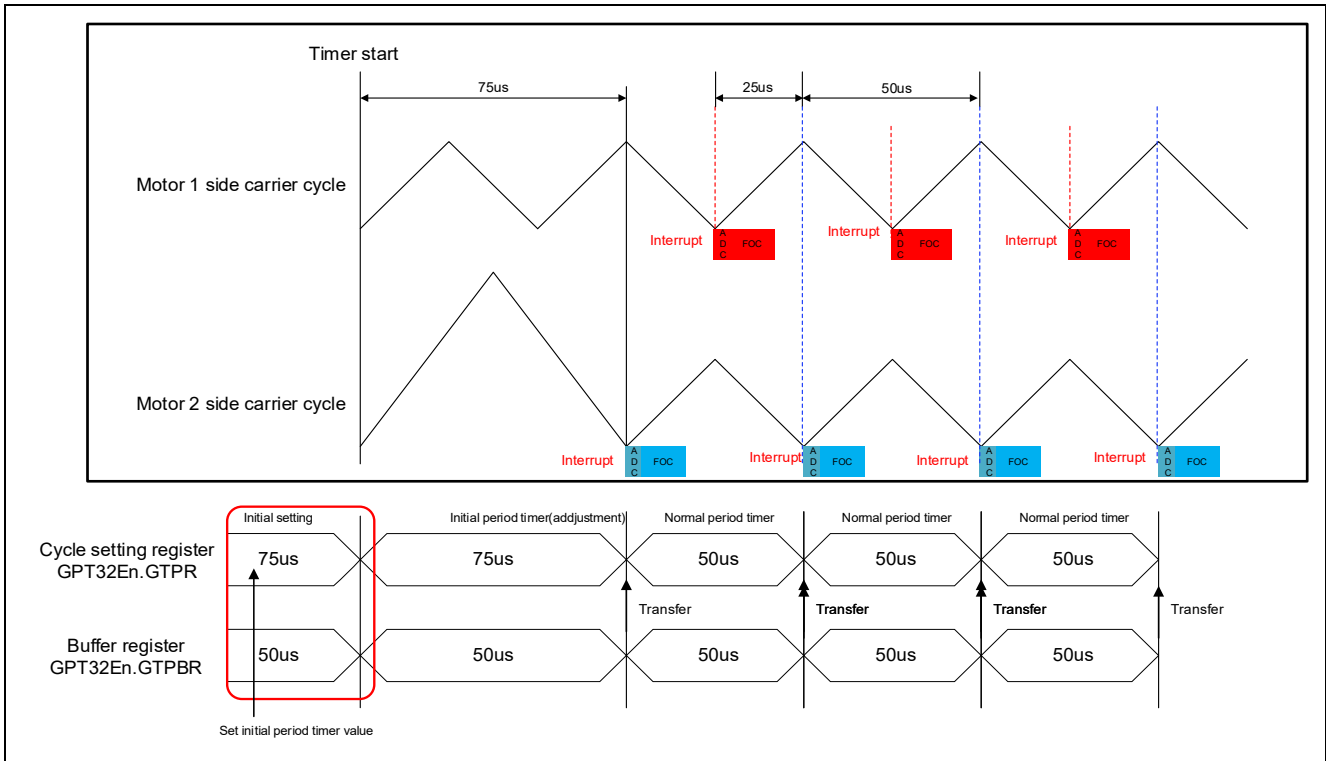
Figure 3-3 Start carrier output

By this setting, the two motors all have 100μsec carrier period. And, carrier synchronized interrupt will not occur at the same time. Interrupt timing is set to each one time skip. Therefore, carrier synchronous interrupts of Motor 1 and Motor 2 occur alternately at periods of 100 [μsec]. It is necessary to implement the processing time of interrupt (including processing time at error occurrence) within 100[μsec].

3.2 Function Specifications of Sensorless Vector Control Software

The block diagram of sensorless vector control is shown below.

In this system, the control cycle of Motor 1 and Motor 2 is set to 100[µs]. The PWM frequency and the control cycle frequency of Motor 1 and Motor 2 should be same due to interrupt timing.
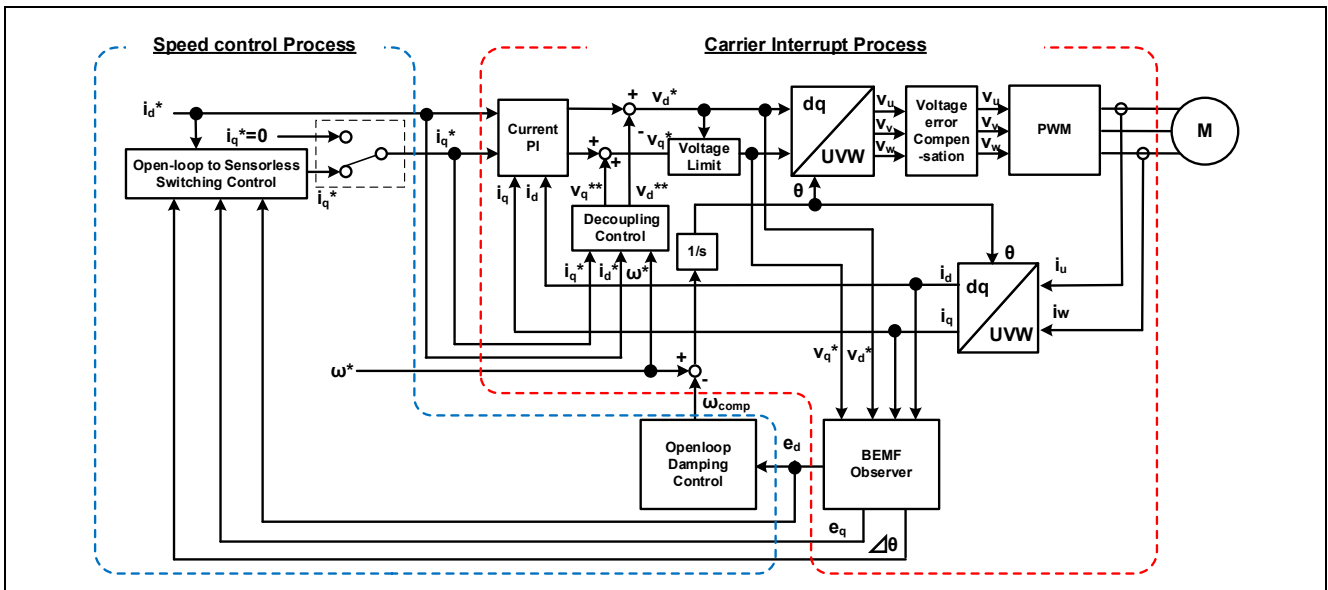
Figure 3-4 Block diagram of sensorless vector control (open-loop control)
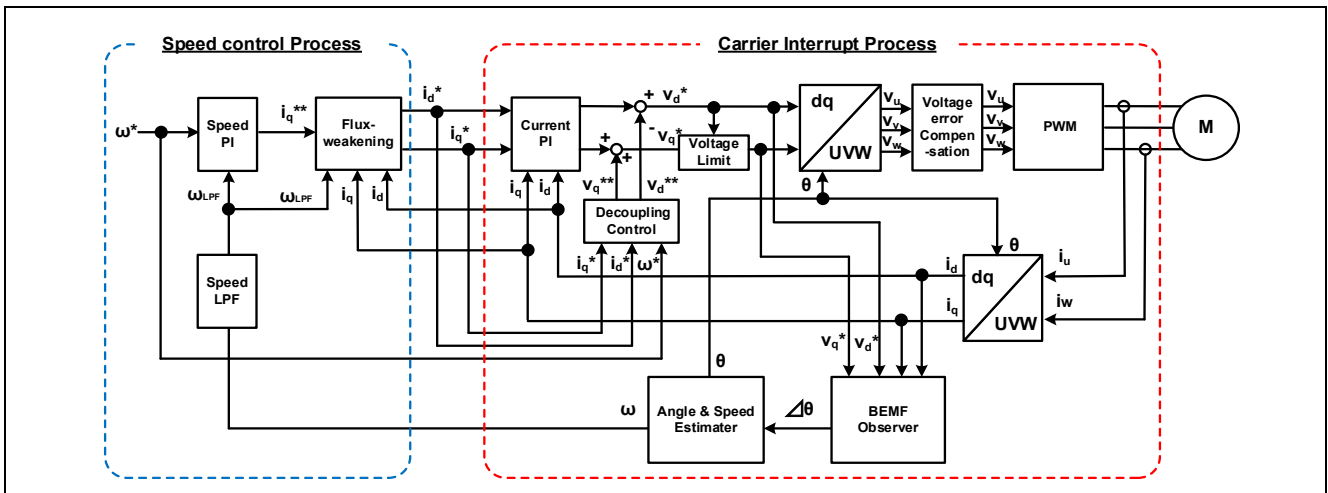
Figure 3-5 Block diagram of sensorless vector control (sensorless control)

3.3 List of functions

Table 3-6 List of functions executed in current control interrupt (1/6)

File name	Function name	Process overview
mtr_main.c	mtr_callback_event0 Input : (motor_callback_args_t *) p_args / Callback argument Output : None	Sensorless vector control callback function for motor1
	mtr_callback_event1 Input : (motor_callback_args_t *) p_args / Callback argument Output : None	Sensorless vector control callback function for motor2
rm_motor_sensorless.c	rm_motor_sensorless_current_callback Input :(motor_current_callback_args_t *) p_args / Callback argument Output : None	Set the speed control output to the current control input
	RM_MOTOR_SENSORLESS_ErrorCheck Input :(motor_ctrl_t * const) p_ctrl / Pointer to control structure (uint16_t * const) p_error / Pointer to get occurred error Output : fsp_err_t / Execution result	Check the occurrence of error
	rm_motor_sensorless_copy_speed_current Input :(motor_speed_output_t *) st_output / Speed control output (motor_current_input_t *) st_input / Current control input Output : None	Copy speed output data to current input data

Table 3-7 List of functions executed in current control interrupt (2/6)

File name	Function name	Process overview
rm_motor_driver.c	rm_motor_driver_cyclic Input :(adc_callback_args_t *) p_args / Callback argument Output : None	Motor driver callback function
	rm_motor_driver_current_get Input :(motor_driver_instance_ctrl_t *) p_ctrl / Pointer to motor driver instance Output : None	Get A/D converted data (phase current & main line voltage)
	RM_MOTOR_DRIVER_FlagCurrentOffsetGet Input :(motor_driver_ctrl_t * const) p_ctrl / Pointer to control structure (uint8_t * const) p_flag_offset / Flag of finish current offset detection Output : fsp_err_t / Execution result	Measure current offset values
	RM_MOTOR_DRIVER_PhaseVoltageSet Input : (motor_driver_ctrl_t * const) p_ctrl / Pointer to control structure (float const) u_voltage / U phase voltage (float const) v_voltage / V phase voltage (float const) w_voltage / W phase voltage Output : fsp_err_t / Execution result	Set phase voltage data to calculate PWM duty.
	rm_motor_driver_modulation Input : (motor_driver_instance_ctrl_t *) p_ctrl / Pointer to motor driver instance Output : None	Perform PWM modulation
	rm_motor_driver_mod_run Input : (motor_driver_modulation_t *) p_mod / Pointer to modulation data structure (const float *) p_f4_v_in / Pointer to 3-phase input voltage (float *) p_f4_duty_out / Where to store the 3-phase output duty cycle Output : None	Calculates duty cycle from input 3-phase voltage (bipolar)
	rm_motor_driver_set_uvw_duty Input : (motor_driver_instance_ctrl_t *) p_ctrl / Pointer to motor driver instance (float) f_duty_u / Duty cycle of phase-U (float) f_duty_v / Duty cycle of phase-V (float) f_duty_w / Duty cycle of phase-W Output : fsp_err_t / Execution result	PWM duty setting
	RM_MOTOR_DRIVER_CurrentGet Input : (motor_driver_ctrl_t * const) p_ctrl / Pointer to control structure (motor_driver_current_get_t * const) p_current_get / Pointer to get data structure Output : fsp_err_t / Execution result	Get calculated phase current, Vdc & Va_max data

Table 3-8 List of functions executed in current control interrupt (3/6)

File name	Function name	Process overview
rm_motor_current.c	rm_motor_current_cyclic Input : (motor_driver_callback_args_t *) p_args / Callback argument Output : None	Current control cycle operation
	RM_MOTOR_CURRENT_ParameterSet Input : (motor_current_ctrl_t * const) p_ctrl / Pointer to control structure (motor_current_input_current_t const * const) p_st_input / Pointer to input current structure Output : fsp_err_t / Execution result	Set (input) parameter data.
	RM_MOTOR_CURRENT_CurrentSet Input : (motor_current_ctrl_t * const) p_ctrl / Pointer to control structure (motor_current_input_current_t const * const) p_st_current / Pointer to input current structure (motor_current_input_voltage_t const * const) p_st_voltage / Pointer to input voltage structure Output : fsp_err_t / Execution result	Set d/q-axis current & voltage data.
	RM_MOTOR_CURRENT_CurrentGet Input : (motor_current_ctrl_t * const) p_ctrl / Pointer to control structure (float * const) p_id / Pointer to get d-axis current (float * const) p_iq / Pointer to get q-axis current Output : fsp_err_t / Execution result	Get d/q-axis current
	motor_current_transform_uvw_dq_abs Input : (const float) f_angle / Rotor angle (const float *) f_uvw / Pointer to UVW-phase array in [U,V,W] format (float *) f_dq / Where to store [d,q] formated array on dq coordinates Output : None	Coordinate transform UVW to dq (absolute transform)

Table 3-9 List of functions executed in current control interrupt (4/6)

File name	Function name	Process overview
	motor_current_angle_cyclic Input : (motor_current_instance_t *) p_instance / Pointer to current control module control instance Output : None	Angle/speed process in cyclic process of current control
	RM_MOTOR_CURRENT_SpeedPhaseSet Input : (motor_current_ctrl_t * const) p_ctrl / Pointer to control structure (float const) speed_rad / Rotational speed (float const) phase_rad / Rotor phase Output : fsp_err_t / Execution result	Set current speed & rotor phase data
	RM_MOTOR_CURRENT_CurrentReferenceSet Input : (motor_current_ctrl_t * const) p_ctrl / Pointer to control structure (float const) speed_rad / D-axis current Reference (float const) phase_rad / Q-axis current Reference Output : fsp_err_t / Execution result	Set current reference data
	RM_MOTOR_CURRENT_PhaseVoltageGet Input : (motor_current_ctrl_t * const) p_ctrl / Pointer to control structure (motor_current_get_voltage_t * const) p_voltage / Pointer to get voltages Output : fsp_err_t / Execution result	Gets the set phase voltage
rm_motor_current.c	motor_current_pi_calculation Input : (motor_current_instance_ctrl_t *) p_instance / Pointer to FOC current control structure Output : None	Calculates output voltage vector from current vector command and actual current vector
	motor_current_pi_control Input : (motor_current_pi_params_t *) pi_ctrl / Pointer to PI control structure Output : float / PI control output value	PI control
	motor_current_limit_abs Input : (float) f4_value / Target value (float) f4_limit_value / Limit Output : float / Limited value	Limit with absolute value
	motor_current_decoupling Input : (motor_current_instance_ctrl_t *) p_ctrl / `Pointer to FOC current control instance (float) f_speed_rad / Electrical speed (const motor_current_motor_parameter_t *) p_mtr / Pointer to motor parameter data structure Output : None	Decoupling control
	motor_current_voltage_limit Input : (motor_current_instance_ctrl_t *) p_ctrl / Pointer to FOC current control structure Output : None	Limit voltage vector

Table 3-10 List of functions executed in current control interrupt (5/6)

File name	Function name	Process overview
rm_motor_current.c	motor_current_transform_dq_uvw_abs Input : (const float) f_angle / Rotor angle (const float *) f_dq / Pointer to dq-axis value array in [D,Q] format (float *) f_uvw / Where to store [U,V,W] formatted 3-phase quantities array Output : None	Coordinate transform dq to UVW 3-phase (absolute transform)
librm_motor_current.a	rm_motor_voltage_error_compensation_main Input : (motor_currnt_voltage_compensation_t *) st_volt_comp / Voltage error compensation data (float *) p_f4_v_array / Reference voltage (float *) p_f4_i_array / Reference current (float) f4_vdc / Bus voltage Output : None	Voltage error compensation
rm_motor_estimate.c	RM_MOTOR_ESTIMATE_FlagPiCtrlSet Input : (motor_angle_ctrl_t * const) p_ctrl / Pointer to control structure (uint32_t const) flag_pi / Flag of PI control runs Output : fsp_err_t / Execution result	Set the flag of PI Control runs
	RM_MOTOR_ESTIMATE_SpeedSet Input : (motor_angle_ctrl_t * const) p_ctrl / Pointer to control structure (float const) speed_ctrl / Control reference of rotational speed (float const) damp_speed / Damping rotational speed Output : fsp_err_t / Execution result	Set speed Information
	RM_MOTOR_ESTIMATE_CurrentSet Input : (motor_angle_ctrl_t * const) p_ctrl / Pointer to control structure (motor_angle_current_t * const) p_st_current / Pointer to current structure (motor_angle_voltage_reference_t * const) p_st_voltage / Pointer to voltage reference structure Output : fsp_err_t / Execution result	Set d/q-axis current data & voltage reference
	RM_MOTOR_ESTIMATE_AngleSpeedGet Input : (motor_angle_ctrl_t * const) p_ctrl / Pointer to control structure (float * const) p_angle / Memory address to get rotor angle data (float * const) p_speed / Memory address to get rotational speed data (float * const) p_phase_err / Memory address to get phase (angle) error data Output : fsp_err_t / Execution result	Gets the current rotor's angle and rotation speed
	RM_MOTOR_ESTIMATE_EstimatedComponentGet Input : (motor_angle_ctrl_t * const) p_ctrl / Pointer to control structure (float * const) p_ed / Memory address to get estimated d-axis component (float * const) p_eq / Memory address to get estimated q-axis component Output : fsp_err_t / Execution result	Gets estimated d/q-axis component

Table 3-11 List of functions executed in current control interrupt (6/6)

File name	Function name	Process overview
r_gpt_three_phase.c	R_GPT_THREE_PHASE_DutyCycleSet Input : (three_phase_ctrl_t * const) p_ctrl / pointer of structure of three phase module variables (three_phase_duty_cycle_t * const) p_duty_cycle / Duty cycle values for all three timer channels Output : fsp_err_t / Execution result	Sets duty cycle for all three timers

Table 3-12 List of functions executed in speed control interrupt (1/3)

File name	Function name	Process overview
mtr_main.c	mtr_callback_event0 Input : motor_callback_args_t * p_args / Callback argument Output : None	Sensorless vector control callback function for motor1
	mtr_callback_event1 Input : motor_callback_args_t * p_args / Callback argument Output : None	Sensorless vector control callback function for motor2
	get_vr1 Input : None Output : uint16_t / Conversion value	Get VR1 A/D conversion value
rm_motor_current.c	RM_MOTOR_CURRENT_ParameterGet Input : (motor_current_ctrl_t * const) p_ctrl / Pointer to control structure (motor_current_input_current_t const * const) p_st_input / Pointer to input current data Output : fsp_err_t / Execution result	Set (input) parameter data
rm_motor_sensorless.c	rm_motor_sensorless_speed_callback Input : (motor_speed_callback_args_t *) p_args / Callback argument Output : None	Speed control callback function
	rm_motor_sensorless_copy_current_speed Input : (motor_current_output_t *) st_output / Pointer to structure of current control output (motor_speed_input_t *) st_input / Pointer to structure of speed control input Output : None	Copy current output data to speed input data

RENESAS

Table 3-13 List of functions executed in speed control interrupt (2/3)

File name	Function name	Process overview
rm_motor_speed.c	rm_motor_speed_cyclic Input : (timer_callback_args_t *) p_args / Callback argument Output : None	Cyclic process of speed control (Call at timer interrupt)
	RM_MOTOR_SPEED_ParameterSet Input : (motor_speed_ctrl_t * const) p_ctrl / Pointer to control structure (motor_speed_input_t const * const) p_st_input / Pointer to structure of speed input parameters Output : fsp_err_t / Execution result	Set speed Input parameters
	RM_MOTOR_SPEED_SpeedControl Input : (motor_speed_ctrl_t * const) p_ctrl / Pointer to control structure Output : fsp_err_t / Execution result	Calculates the d/q-axis current reference. (Main process of Speed Control)
	rm_motor_speed_set_speed_ref Input : (motor_speed_instance_ctrl_t *) p_ctrl / Pointer to FOC data instance Output : float / Reference speed	Updates the speed reference
	rm_motor_speed_set_iq_ref Input : (motor_speed_instance_ctrl_t *) p_ctrl / Pointer to control instance Output : float / Iq reference	Updates the q-axis current reference
	rm_motor_speed_set_id_ref Input : (motor_speed_instance_ctrl_t *) p_ctrl / Pointer to control instance Output : float / Id reference	Updates the d-axis current reference
	RM_MOTOR_SPEED_ParameterGet Input : (motor_speed_ctrl_t * const) p_ctrl / Pointer to motor speed control block (motor_speed_output_t * const) p_st_output / Pointer to get speed control parameters Output : fsp_err_t / Execution result	Get speed control output parameters

RENESAS

Table 3-14 List of functions executed in speed control interrupt (3/3)

File name	Function name	Process overview
librm_motor_speed.a	rm_motor_speed_first_order_lpf Input : (motor_speed_lpf_t *) p_lpf / Pointer to first order LPF structure (float) f_input / Input data Output : float / Filtered data	First order LPF
	rm_motor_speed_fluxwkn_set_vamax Input : (motor_speed_flux_weakening_t *) p_fluxwkn / Pointer to flux weakening structure (float) f4_va_max / maximum magnitude of voltage vector Output : None	Sets the maximum magnitude of voltage vector
	rm_motor_speed_fluxwkn_run Input : (motor_speed_flux_weakening_t *) p_fluxwkn / Pointer to flux weakening structure (float) f4_speed_rad / Electrical speed of motor (const float *) p_f4_idq / Pointer to the measured current vector in format d/q (float *) p_f4_idq_ref / Pointer to reference current vector in format d/q Output : None	Executes the flux-weakening module

3.4 Contents of Control

3.4.1 Configuration options

The configuration options of the sensorless vector control module can be configured using the RA Configurator. The changed options are automatically reflected to the hal_data.h and rm_motor_sensorless.h when generating code. The option names and setting values are listed in the shown as follows. The settings are not different between motor1 and 2.

Table 3-15 Configuration options(rm_motor_sensorless.h)

Options	Description
Limit of over current (A)	When a phase current exceeds this value, PWM output ports are set to off.
Limit of over voltage (V)	When an inverter voltage exceeds this value, PWM output ports are set to off.
Limit of over speed (rpm)	When a rotation speed exceeds this value, PWM output ports are set to off.
Limit of low voltage (V)	When an inverter voltage becomes below this value, PWM output ports are set to off.

Table 3-16 Configuration options(rm_motor_sensorless.h)

Options	RA6T1
Limit of over current (A)	0.42
Limit of over voltage (V)	28.0
Limit of over speed (rpm)	3000.0
Limit of low voltage (A)	14.0

3.4.2 Configuration Options for included modules

The sensorless vector control module for motor includes below modules.

- Current Module
- Speed Module
- Angle Module
- Driver Module

And also these included modules have each configuration parameters as same as the sensorless vector control module. The option names and setting values are listed in the tables shown as follows.

The settings are not different between motor 1 and 2 except driver module.

Table 3-17 Configuration options (rm_motor_current.h)

Options	Description
Shunt type	Selects how many shunt resistances to use current detection.
Current control decimation	Counts of decimation about carrier interrupt
Input voltage (V)	Input voltage [V]
Sample delay compensation	Selects whether to "enable" or "disable" sample delay compensation
Voltage error compensation	Selects whether to "enable" or "disable" voltage error compensation.
Voltage error compensation table of voltage 1	Table of voltage error compensation about voltage #1
Voltage error compensation table of voltage 2	Table of voltage error compensation about voltage #2
Voltage error compensation table of voltage 3	Table of voltage error compensation about voltage #3
Voltage error compensation table of voltage 4	Table of voltage error compensation about voltage #4
Voltage error compensation table of voltage 5	Table of voltage error compensation about voltage #5
Voltage error compensation table of current 1	Table of voltage error compensation about current #1
Voltage error compensation table of current 2	Table of voltage error compensation about current #2
Voltage error compensation table of current 3	Table of voltage error compensation about current #3
Voltage error compensation table of current 4	Table of voltage error compensation about current #4
Voltage error compensation table of current 5	Table of voltage error compensation about current #5
Design Parameter | Current PI loop omega	Current PI control omega parameter [Hz].
Design Parameter | Current PI loop zeta	Current PI control zeta parameter.
Motor Parameter | Pole pairs	Pole pairs of target motor.
Motor Parameter | Resistance (ohm)	Resistance of motor [ohm].
Motor Parameter | Inductance of d-axis (H)	D-axis inductance [H].
Motor Parameter | Inductance of q-axis (H)	Q-axis inductance [H].
Motor Parameter | Permanent magnetic flux (Wb)	Magnetic flux [Wb].
Motor Parameter | Rotor inertia (kgm^2)	Rotor inertia [kgm^2].

Table 3-18 Configuration Options initial value (rm_motor_current.h)

Options	RA6T1
Shunt type	2shunt
Current control decimation	1
PWM carrier frequency (kHz)	20.0
Input voltage (V)	24.0
Sample delay compensation	Disable
Voltage error compensation	Enable
Voltage error compensation table of voltage 1	0.672
Voltage error compensation table of voltage 2	0.945
Voltage error compensation table of voltage 3	1.054
Voltage error compensation table of voltage 4	1.109
Voltage error compensation table of voltage 5	1.192
Voltage error compensation table of current 1	0.013
Voltage error compensation table of current 2	0.049
Voltage error compensation table of current 3	0.080
Voltage error compensation table of current 4	0.184
Voltage error compensation table of current 5	0.751
Design Parameter | Current PI loop omega	300.0
Design Parameter | Current PI loop zeta	1.0
Motor Parameter | Pole pairs	2U
Motor Parameter | Resistance (ohm)	8.5F
Motor Parameter | Inductance of d-axis (H)	0.0045F
Motor Parameter | Inductance of q-axis (H)	0.0045F
Motor Parameter | Permanent magnetic flux (Wb)	0.02159F
Motor Parameter | Rotor inertia (kgm^2)	0.0000028F

Table 3-19 Configuration options (rm_motor_speed.h) (1/2)

Options	Description
Speed control period (sec)	The period of speed control process [sec].
Step of speed climbing (rpm)	The step of speed fluctuation [rpm]. Program controls speed by this step at acceleration and deceleration.
Maximum rotational speed (rpm)	Maximum rotational speed [rpm]
Speed LPF omega	Speed LPF parameter omega [Hz].
Limit of q-axis current (A)	Limit of q-axis current [A].
Step of speed feedback at open-loop	Rate of reference speed for feedback speed limiter at Open-Loop.
Natural frequency	
Open-loop damping	Select enable/disable of damping control at Open-Loop.
Flux weakening	Select enable/disable of flux weakening control at high speed.
Torque compensation for sensorless transition	Select enable/disable of soft switching at the transition from Open-Loop to PI control.
Speed observer	Select enable/disable of speed observer process
Selection of speed observer	Select the method of speed observer

Table 3-20 Configuration options (rm_motor_speed.h) (2/2)

Options	Description
Open-Loop	Step of d-axis current climbing	The d-axis current reference ramping up rate [A/msec].
Open-Loop	Step of d-axis current descending	The d-axis current reference ramping down rate [A/msec].
Open-Loop	Step of q-axis current descending ratio	The q-axis current reference ramping down proportion to reference before open-loop [A/msec].
Open-Loop	Reference of d-axis current	The d-axis current reference in open-loop drive [A].
Open-Loop	Threshold of speed control descending	The speed threshold [rad/s] to ramp down the d-axis current [rpm].
Open-Loop	Threshold of speed control climbing	The speed threshold [rad/s] to ramp up the d-axis current [rpm].
Open-Loop	Period between open-loop to BEMF (sec)	Time to switch open-loop to sensor-less [sec].
Open-Loop	Phase error(degree) to decide sensor-less switch timing	Phase error to decide sensor-less switch timing (electrical angle) [degree].
Design parameter	Speed PI loop omega	Speed PI Control parameter omega.
Design parameter	Speed PI loop zeta	Speed PI Control parameter zeta.
Design parameter	Estimated d-axis HPF omega	Natural frequency [Hz] for HPF in open-loop damping gain design.
Design parameter	Open-loop damping zeta	Damping ratio for open-loop damping gain design.
Design parameter	Cutoff frequency of phase error LPF	The cut-off frequency [Hz] of phase error LPF gain design.
Design parameter	Speed observer omega	Speed observer omega.
Design parameter	Speed observer zeta	Speed observer zeta.
Motor Parameter	Pole pairs	Pole pairs of target motor.
Motor Parameter	Resistance (ohm)	Resistance of motor [ohm].
Motor Parameter	Inductance of d-axis (H)	D-axis inductance [H].
Motor Parameter	Inductance of q-axis (H)	Q-axis inductance [H].
Motor Parameter	Permanent magnetic flux (Wb)	Magnetic flux [Wb].
Motor Parameter	Rotor inertia (kgm^2)	Rotor inertia [kgm^2].

Table 3-21 Configuration Options initial value (rm_motor_speed.h)

Options	RA6T1
Speed control period (sec)	0.001
Step of speed climbing (rpm)	0.5
Maximum rotational speed (rpm)	2650.0F
Speed LPF omega	10.0
Speed at Id climbing (rpm)	500.0F
Limit of q-axis current (A)	0.42F
Step of speed feedback at open-loop	0.2F
Open-loop damping	Enable
Flux weakening	Disable
Torque compensation for sensorless transition	Enable
Speed observer	Enable
Selection of speed observer	Disturbance
Open-Loop | Step of d-axis current climbing	0.3F
Open-Loop | Step of d-axis current descending	0.3F
Open-Loop | Step of q-axis current descending ratio	1.0F
Open-Loop | Reference of d-axis current	0.3F
Open-Loop | Threshold of speed control descending	600.0F
Open-Loop | Threshold of speed control climbing	500.0F
Open-Loop | Period between open-loop to BEMF (sec)	0.025F
Open-Loop | Phase error(degree) to decide sensorless switch timing	10.0F
Design parameter | Speed PI loop omega	5.0F
Design parameter | Speed PI loop zeta	1.0F
Design parameter | Estimated d-axis HPF omega	2.5F
Design parameter | Open-loop damping zeta	1.0F
Design parameter | Cutoff frequency of phase error LPF	10.0F
Design parameter | Speed observer omega	200.0
Design parameter | Speed observer zeta	1.0
Motor Parameter | Pole pairs	2U
Motor Parameter | Resistance (ohm)	8.5F
Motor Parameter | Inductance of d-axis (H)	0.0045F
Motor Parameter | Inductance of q-axis (H)	0.0045F
Motor Parameter | Permanent magnetic flux (Wb)	0.02159F
Motor Parameter | Rotor inertia (kgm^2)	0.0000028F

Table 3-22 Configuration options (rm_motor_estimate.h)

Options	Description
Motor Parameter	Pole pairs	Pole pairs of target motor.
Motor Parameter	Resistance (ohm)	Resistance of motor [ohm].
Motor Parameter	Inductance of d-axis (H)	D-axis inductance [H].
Motor Parameter	Inductance of q-axis (H)	Q-axis inductance [H].
Motor Parameter	Permanent magnetic flux (Wb)	Magnetic flux [Wb].
Motor Parameter	Rotor inertia (kgm^2)	Rotor inertia [kgm^2].
Openloop damping	Select enable/disable of Open-Loop Damping Control
Natural frequency of BEMF observer	Natural frequency for BEMF observer [Hz].
Damping ratio of BEMF observer	Damping ratio for BEMF observer.
Natural frequency of PLL Speed estimate loop	Natural frequency for rotor position Phase-Locked Loop [Hz].
Damping ratio of PLL Speed estimate loop	Damping ratio for rotor position Phase-Locked Loop.
Control period	Period of Speed Control [sec]

Table 3-23 Configuration Options initial value (rm_motor_estimate.h)

Options	RA6T1
Motor Parameter	Pole pairs	2U
Motor Parameter	Resistance (ohm)	8.5F
Motor Parameter	Inductance of d-axis (H)	0.0045F
Motor Parameter	Inductance of q-axis (H)	0.0045F
Motor Parameter	Permanent magnetic flux (Wb)	0.02159F
Motor Parameter	Rotor inertia (kgm^2)	0.0000028F
Openloop damping	Enable
Natural frequency of BEMF observer	1000.0
Damping ratio of BEMF observer	1.0
Natural frequency of PLL Speed estimate loop	20.0
Damping ratio of PLL Speed estimate loop	1.0
Control period	0.0001

Table 3-24 Configuration options (rm_motor_driver.h)

Options	Description
Common | Shared ADC support	Selection of using shared ADC module
Common | Supported Motor Number	Number of driven motors
Shunt type	Current detection method selection
Modulation method	Selection of the method of modulation
PWM output port UP	Port setting of U phase upper arm
PWM output port UN	Port setting of U phase lower arm
PWM output port VP	Port setting of V phase upper arm
PWM output port VN	Port setting of V phase lower arm
PWM output port WP	Port setting of W phase upper arm
PWM output port WN	Port setting of W phase lower arm
PWM timer frequency (MHz)	PWM Timer Clock Frequency [MHz]
PWM carrier period (Micro seconds)	PWM Carrier Period [Micro seconds]
Dead time (raw counts)	PWM Dead time [raw counts]
Current range (A)	Measurement Range of Electric current [A]
Voltage range (V)	Measurement Range of Inverter Voltage [V]
Counts for current offset measurement	Counts of measurement the offset of A/D Conversion at electric current input.
A/D conversion channel for U phase current	A/D channel for U-phase current
A/D conversion channel for W phase current	A/D channel for W-phase current
A/D conversion channel for main line voltage	A/D channel for main line voltage
A/D conversion channel for V phase current	A/D channel for V-phase current It is invalid at 2shunt detection.
A/D conversion unit for U Phase current	ADC unit number of U phase current
A/D conversion unit for W Phase current	ADC unit number of W phase current
A/D conversion unit for main line voltage	ADC unit number of main line voltage
A/D conversion unit for V Phase current	ADC unit number of V phase current It is invalid at 2shunt detection.
ADC interrupt module	Select ADC module which occurs interrupt
Input voltage	Range of input for main line voltage
Resolution of A/D conversion	Resolution of A/D conversion Please set same value with ADC module setting.
Offset of A/D conversion for current	Offset level of A/D conversion input for current Please set according to the circuit.
Conversion level of A/D conversion for voltage	Conversion level of A/D conversion for voltage Please set when the CPU main voltage is different.
GTIOCA stop level	Output level of upper arm at stop status
GTIOCB stop level	Output level of lower arm at stop status
Maximum duty	Maximum duty of PWM Maximum duty except dead time.

RENESAS

Table 3-25 Configuration Options initial value (rm_motor_estimate.h)

Options	RA6T1
Common | Shared ADC support	Enabled
Common | Supported Motor Number	2
Shunt type	2shunt
Modulation method	SVPWM
PWM output port UP	motor 1 : BSP_IO_PORT_04_PIN_15
	motor 2 : BSP_IO_PORT_01_PIN_15
PWM output port UN	motor 1 : BSP_IO_PORT_04_PIN_14
	motor 2 : BSP_IO_PORT_06_PIN_08
PWM output port VP	motor 1 : BSP_IO_PORT_01_PIN_13
	motor 2 : BSP_IO_PORT_06_PIN_09
PWM output port VN	motor 1 : BSP_IO_PORT_01_PIN_14
	motor 2 : BSP_IO_PORT_06_PIN_10
PWM output port WP	motor 1 : BSP_IO_PORT_01_PIN_11
	motor 2 : BSP_IO_PORT_06_PIN_01
PWM output port WN	motor 1 : BSP_IO_PORT_01_PIN_12
	motor 2 : BSP_IO_PORT_06_PIN_00
PWM timer frequency (MHz)	120
PWM carrier period (Micro seconds)	50
Dead time (raw counts)	240
Current range (A)	27.5F
Voltage range (V)	111.0F
Counts for current offset measurement	500U
A/D conversion channel for U phase current	motor 1 : 0
	motor 2 : 0
A/D conversion channel for W phase current	motor 1 : 2
	motor 2 : 2
A/D conversion channel for main line voltage	motor 1 : 5
	motor 2 : 3
A/D conversion channel for V phase current	motor 1 : 1
	motor 2 : - (invalid)
A/D conversion unit for U Phase current	motor 1 : 0
	motor 2 : 1
A/D conversion unit for W Phase current	motor 1 : 0
	motor 2 : 1
A/D conversion unit for main line voltage	motor 1 : 0
	motor 2 : 0
A/D conversion unit for V Phase current	motor 1 : 0
	motor 2 : - (invalid)
ADC interrupt module	motor 1 : 1st
	motor 2 : 2nd
Input voltage	24.0
Resolution of A/D conversion	0xFFF
Offset of A/D conversion for current	0x7FF
Conversion level of A/D conversion for voltage	1.0
GTIOCA stop level	Pin Level Low
GTIOCB stop level	Pin Level High
Maximum duty	0.9375

3.5 Control flowcharts
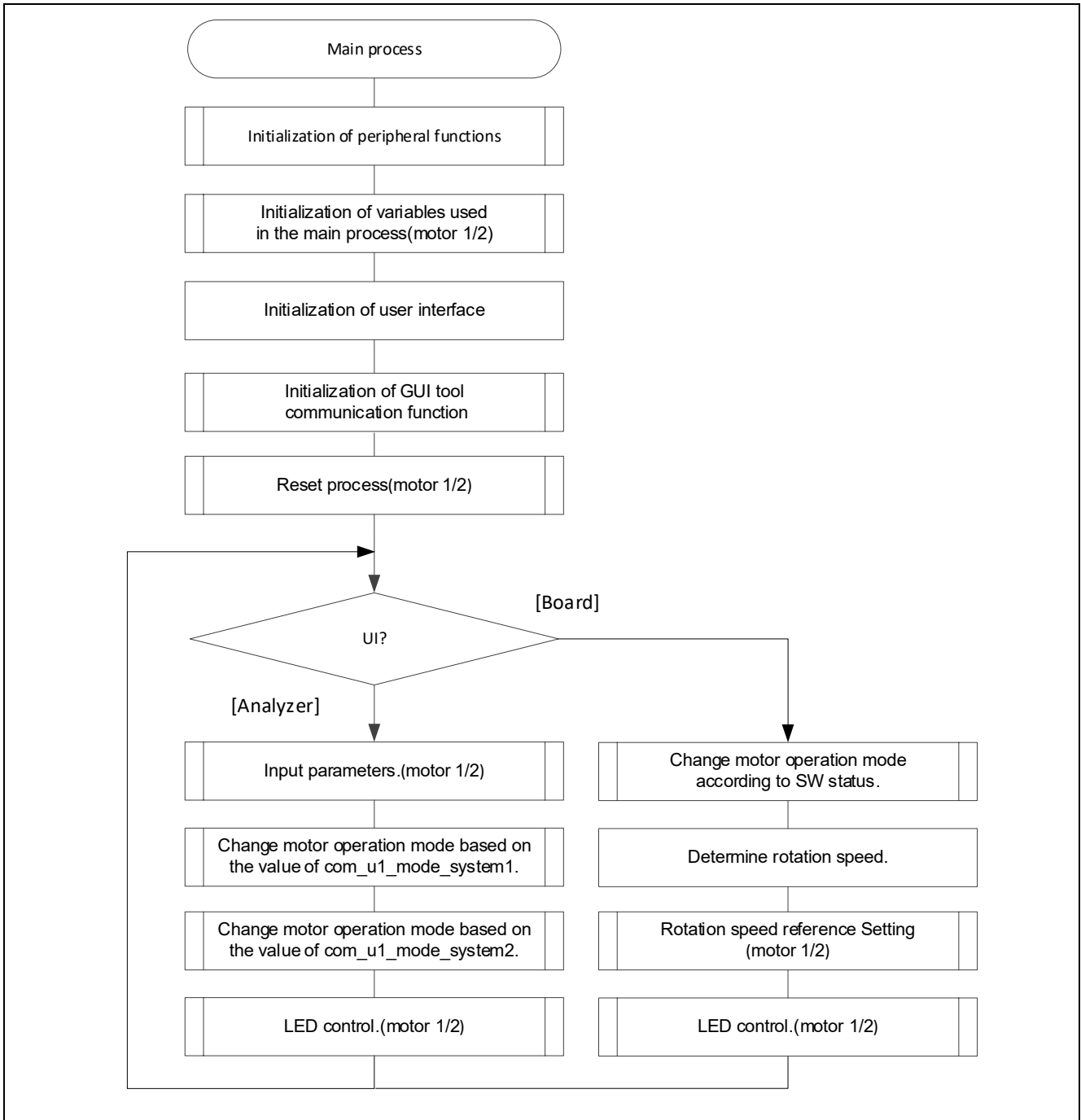
3.5.1 Main process

Figure 3-6 Main process flowchart

3.5.2 Current control period interrupt (carrier synchronized interrupt) process
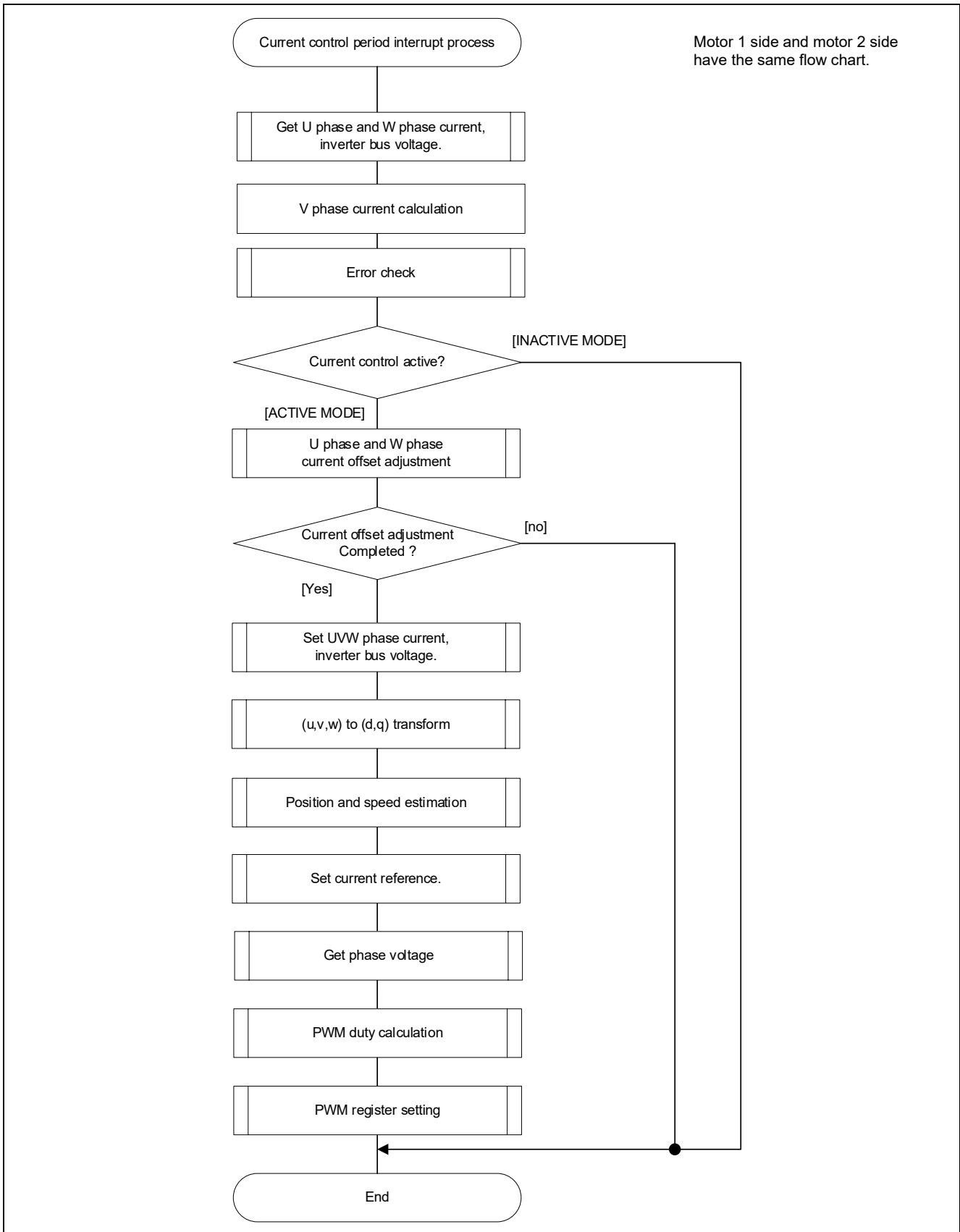

Figure 3-7 Current control Period Interrupt (carrier interrupt) process flowchart
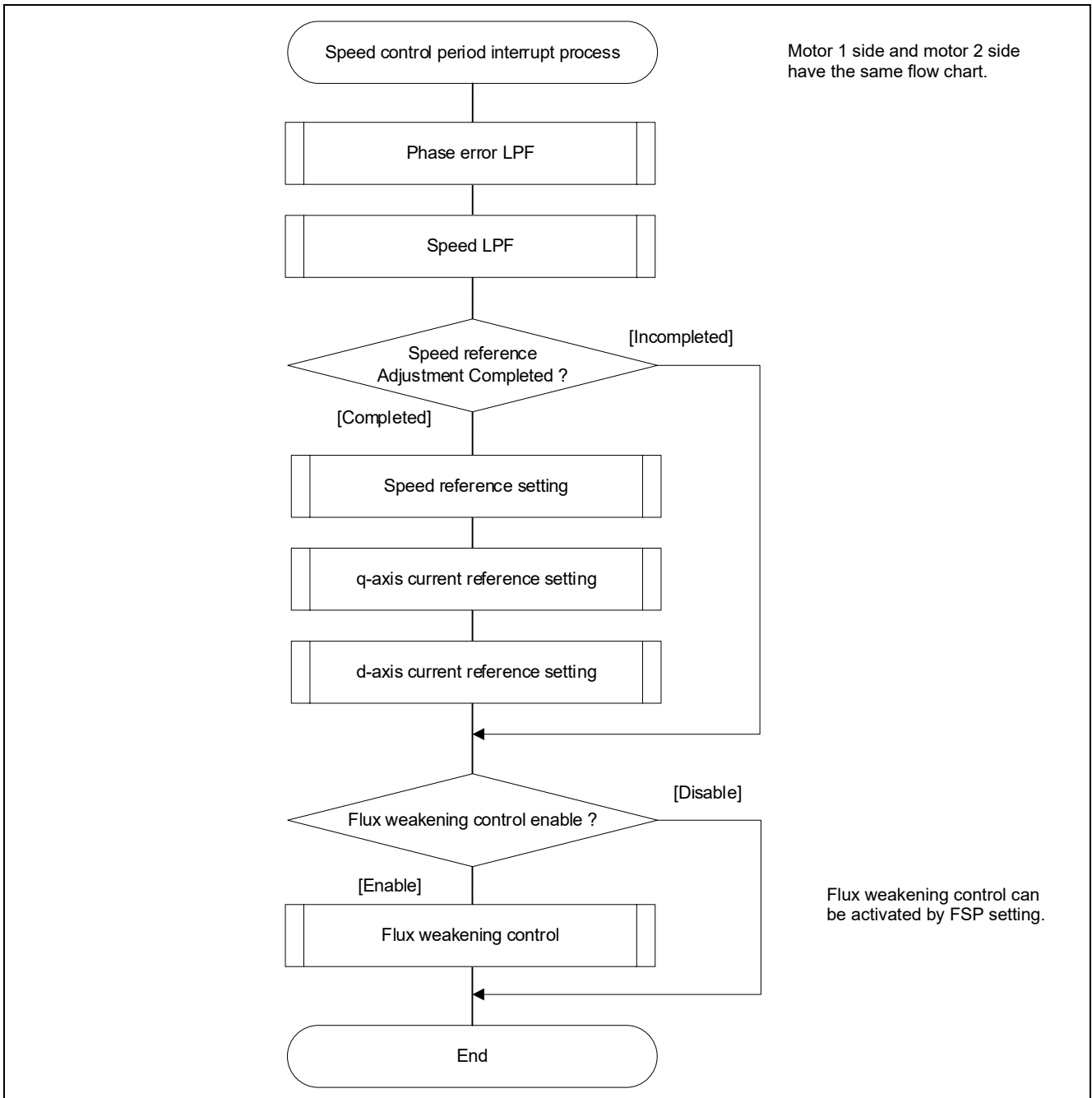
3.5.3　Speed control period interrupt process

Figure 3-8 Speed Control period interrupt process flowchart

3.5.4 Over current detection interrupt process

The overcurrent detection interrupt is an interrupt that occurs when an external overcurrent detection signal is input. The PWM output terminal are put in the high impedance state. Therefore, at the start of execution of this interrupt processing, the PWM output terminal is already in the high impedance state and the output to the motor had been stopped.
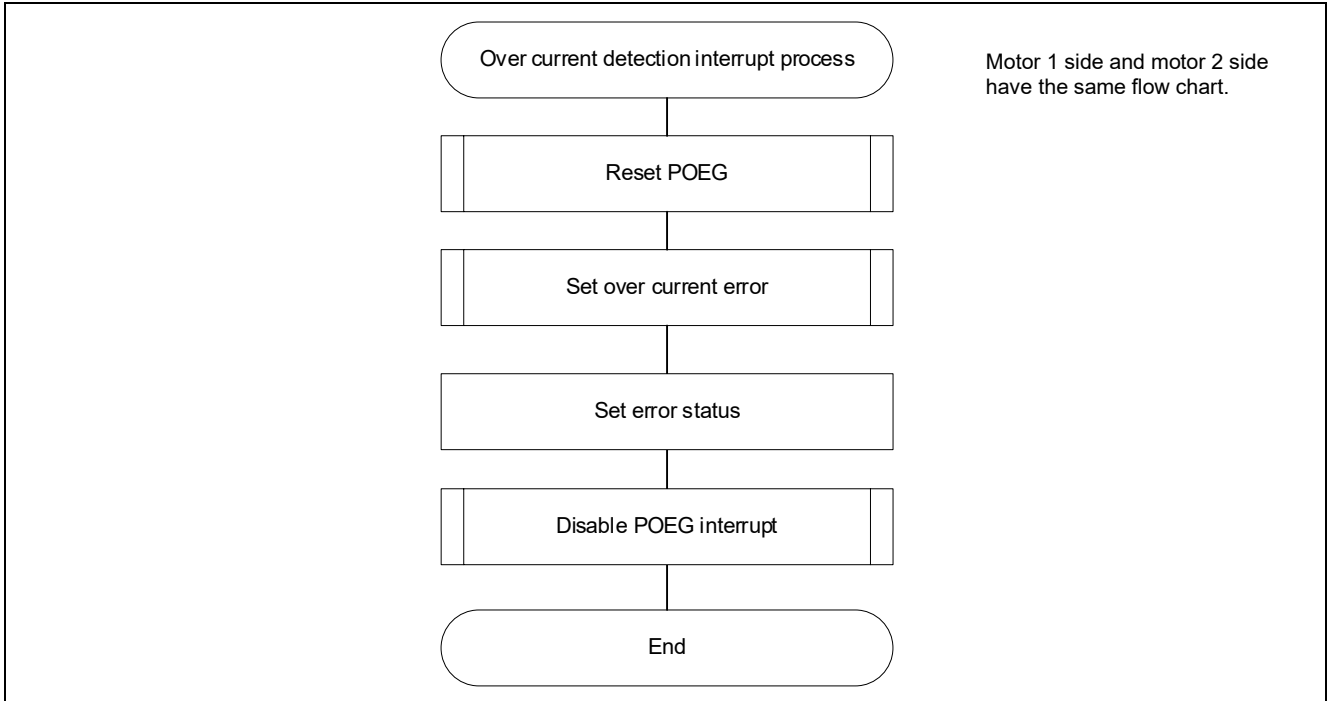
Figure 3-9 Over current detection interrupt process flowchart

4. Project Operation Overview

This section explains the operation of the sample program.

4.1 Quick Start

When executing the sample code only in the evaluation environment without using Renesas motor workbench, the following procedure can be executed

(1) After turning on stabilized power supply or executing reset, LED1_1, LED1_2, LED2_1, and LED2_2 on the inverter board are both off and the motor stops.

(2) IF the toggle switch (SW1_1, SW1_2) on the inverter board is turned on, the motor starts to rotate. Every time the toggle switch (SW1_1, SW1_2) is changed, motor rotation starts/stops alternately. If the motor rotates normally, LED1_1, LED1_2 on the inverter board is on. However, if LED2_1, LED2_2 on the inverter board is also on, error is occurring.

(3) In order to change the direction of the motor rotation, adjust it with the variable resistor (VR1) on the inverter board.
 — Turn the variable resistor (VR1) right: Both motor 1 and motor 2 rotate clockwise
 — Turn the variable resistor (VR1) left: Both motor 1 and motor 2 rotate counterclockwise

(4) If error occurs, LED2_1 or LED2_2 on the inverter board lighten, and the motor rotation stops. To restore, the toggle switch (SW1_1, SW1_2) on the inverter board needs to be turned off, the push switch (SW2_1, SW2_2) to be pushed and released,

(5) In order to stop the operation check, turn off the output of the stabilized power supply after making sure that the motor rotation has already stopped.

4.2 Motor Control Development Support Tool 'Renesas Motor Workbench'

4.2.1 Overview

In the target software of this application note, the motor control development support tool "Renesas Motor Workbench" is used as a user interface (rotating/stop motor, set rotation speed reference, etc). Please refer to 'Renesas Motor Workbench User's Manual' for usage and more details.

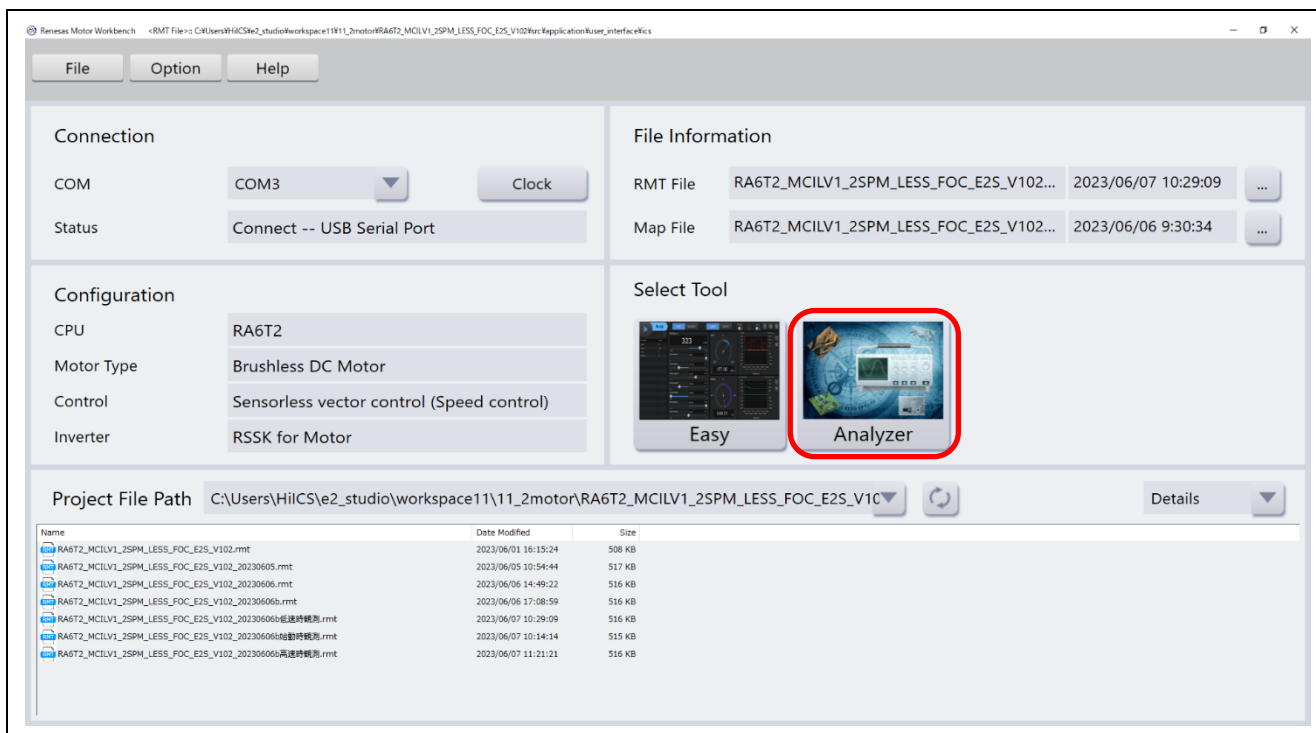You can find 'Renesas Motor Workbench' on Renesas Electronics Corporation website.

Figure 4-1 Renesas Motor Workbench – Appearance

- Set up for 'Renesas Motor Workbench'

(1) Start 'Renesas Motor Workbench' by clicking this icon.
(2) Drop down menu [File] → [Open RMT File(O)].
   And select RMT file in '[Project Folder]/src/application/user_interface/ics/'.
(3) Use the 'Connection' [COM] select menu to choose the COM port.
(4) Click the Analyzer button of Select Tool to activate Analyzer function.
(5) Please refer to '4.2.3 Operation example for Analyzer' for motor driving operation.

4.2.2 List of variables for Analyzer function

Table 4-1 is a list of variables for Analyzer. These variables are reflected to the corresponding variables when the same values as g_u1_enable_write are written to com_u1_enable_write. However, note that variables with (*) do not depend on com_u1_enable_write.

When controlling Motor 1, use variables with "1" in variable name. When controlling Motor 2, use variables with "2" in variable name.

Table 4-1 List of Variables for Analyzer

Variable name	Type	Content
com_u1_sw_userif (*)	uint8_t	User interface switch 0: Analyzer use 1: Board user interface use (default)
com_u1_mode_system(*)	uint8_t	State management 0: Stop mode 1: Run mode 3: Reset
com_f4_ref_speed_rpm	float	Speed reference (mechanical angle) [rpm]
com_u2_mtr_pp	uint16_t	Number of pole pairs
com_f4_mtr_r	float	Resistance [Ω]
com_f4_mtr_ld	float	d-axis Inductance [H]
com_f4_mtr_lq	float	q-axis Inductance [H]
com_f4_mtr_m	float	Magnetic Flux [Wb]
com_f4_mtr_j	float	Inertia [kgm^2]
com_f4_current_omega	float	Natural frequency of current control system [Hz]
com_f4_current_zeta	float	Damping ratio of current control system
com_f4_speed_omega	float	Natural frequency of speed control system [Hz]
com_f4_speed_zeta	float	Damping ratio of speed control system
com_f4_e_obs_omega	float	Natural frequency of BEMF estimation system [Hz]
com_f4_e_obs_zeta	float	Damping ratio of BEMF estimation system
com_f4_pll_est_omega	float	Natural frequency of position estimation system [Hz]
com_f4_pll_est_zeta	float	Damping ratio of position estimation system
com_f4_ref_id	float	d-axis current reference in open loop mode [A]
com_f4_ol_id_up_step	float	d-axis current reference ramping up rate
com_f4_ol_id_down_step	float	d-axis current reference ramping down rate
com_f4_id_down_speed_rpm	float	Speed when start to subtract d-axis current reference (mechanical angle) [rpm]
com_f4_id_up_speed_rpm	float	Speed when start to add d-axis current reference (mechanical angle) [rpm]
com_f4_max_speed_rpm	float	Maximum speed value (mechanical angle) [rpm]
com_f4_overspeed_limit_rpm	float	Speed limit (mechanical angle) [rpm]
com_f4_overcurrent_limit	float	Over current limit [A]
com_f4_iq_limit	float	q-axis current limit [A]
com_f4_limit_speed_change	float	Change speed limit (electrical angle) [rad/s]
com_u1_enable_write	uint8_t	Enabled to rewriting variables (rewritten when the same values as g_u1_enable_write are written)

4.2.3 Operation example for Analyzer

This section shows an example of motor driving operation using the Analyzer. Please refer to Figure 4-2 for operation "Control Window". Regarding the specification of "Control Window", refer to 'Renesas Motor Workbench User's Manual'.

Variables in variable name are for Motor 1 control. When controlling Motor 2, use variables with "1" in variable name. The following operation example is described by the Motor 1 control variables only.

- Change the user interface to Analyzer
  (1) Confirm the checkboxes of column [W?] for 'com_u1_sw_userif' marks.
  (2) Input '0' in the [Write] box of 'com_u1_sw_userif'.
  (3) Click the 'Write' button.

- Driving the motor
  (1) The [W?] check boxes contain checkmarks for "com_u1_mode_system","com_f4_ref_speed_rpm", "com_u1_enable_write"
  (2) Type a reference speed value in the [Write] box of "com_f4_ref_speed_rpm".
  (3) Click the "Write" button.
  (4) Click the "Read" button. Confirm the [Read] box of "com_f4_ref_speed_rpm" , "g_u1_enable_write".
  (5) Type a same value of "g_u1_enable_write" in the [Write] box of "com_u1_enable_write".
  (6) Type a value of "1" in the [Write] box of "com_u1_mode_system".
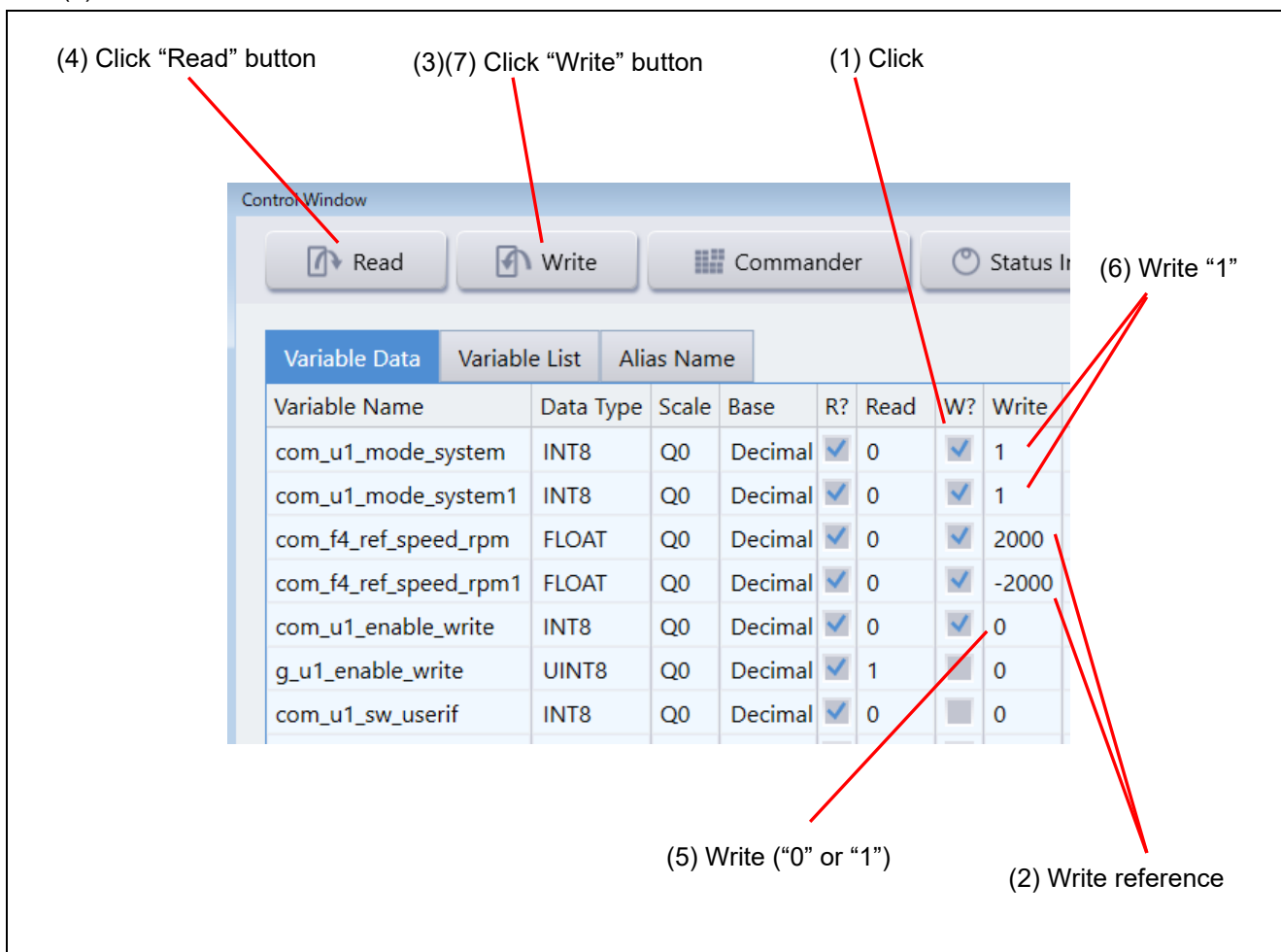 (7) Click the "Write" button.

Figure 4-2 Procedure - Driving the motor

- Stop the motor
  (1) Type a value of "0" in the [Write] box of "com_u1_mode_system".
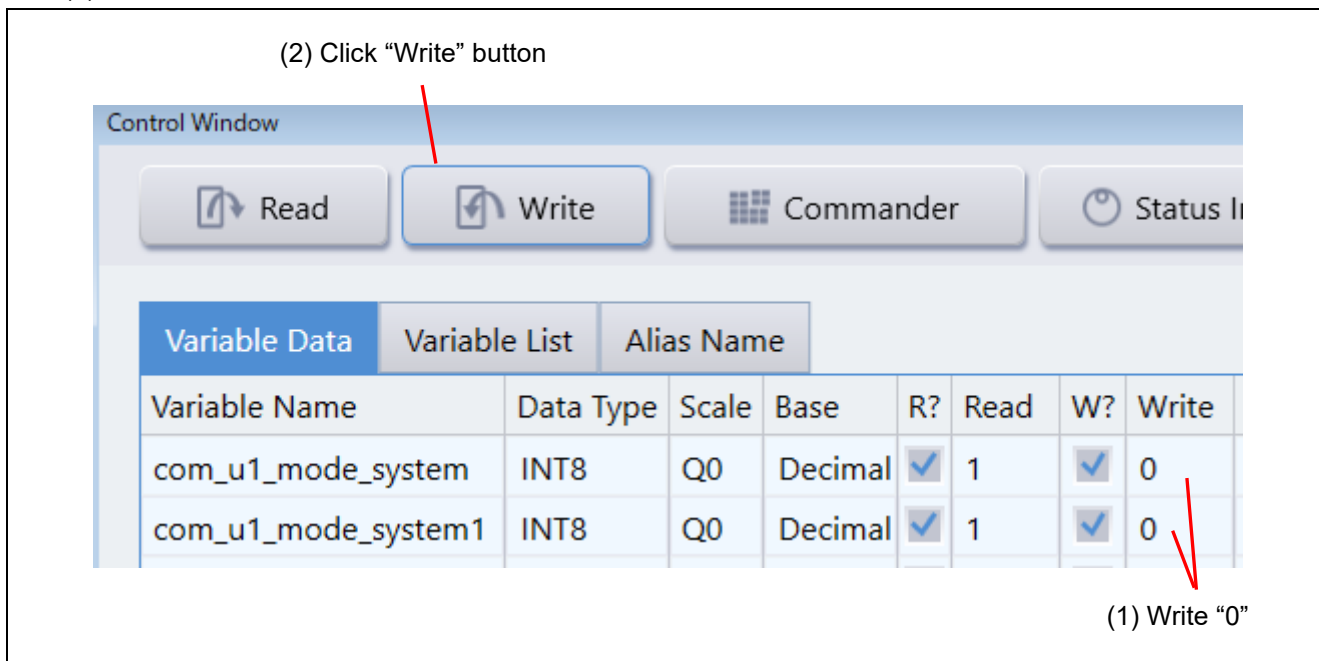 (2) Click the "Write" button.

Figure 4-3 Procedure - Stop the motor

- Error cancel operation
  (1) Type a value of "3" in the [Write] box of "com_u1_mode_system".
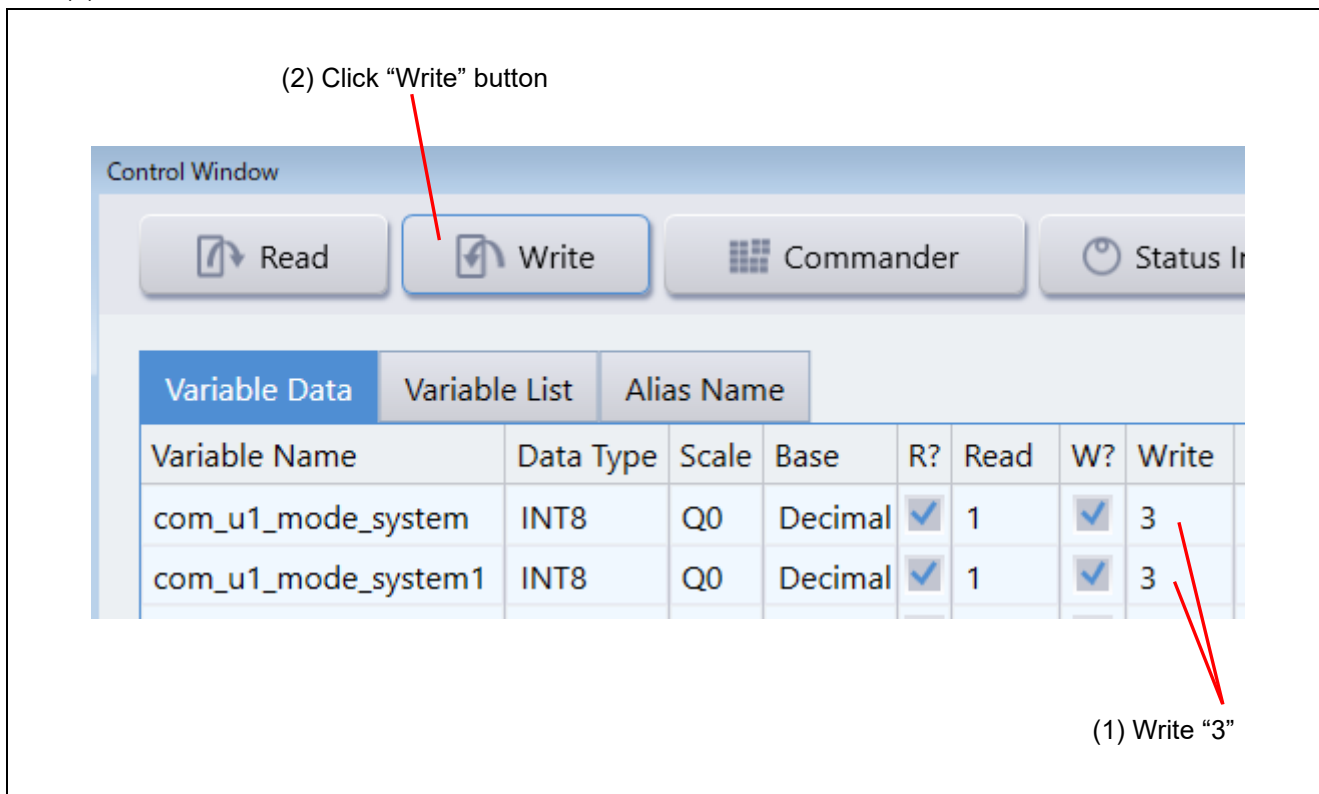 (2) Click the "Write" button.

Figure 4-4 Procedure - Error cancel operation

4.2.4 Example of changing communication speed

The procedure for changing the communication speed of Renesas Motor Workbench with the sample software is shown below. See the Renesas Motor Workbench User's Manual for the values to change.

- Change the communication speed setting of the sample software (when the required communication rate is 10 Mbps)

  (1) Change the value of ICS_BRR in r_mtr_ics.h to 1.

  (2) Change the value of MTR_ICS_DECIMATION in r_mtr_ics.h to 1.
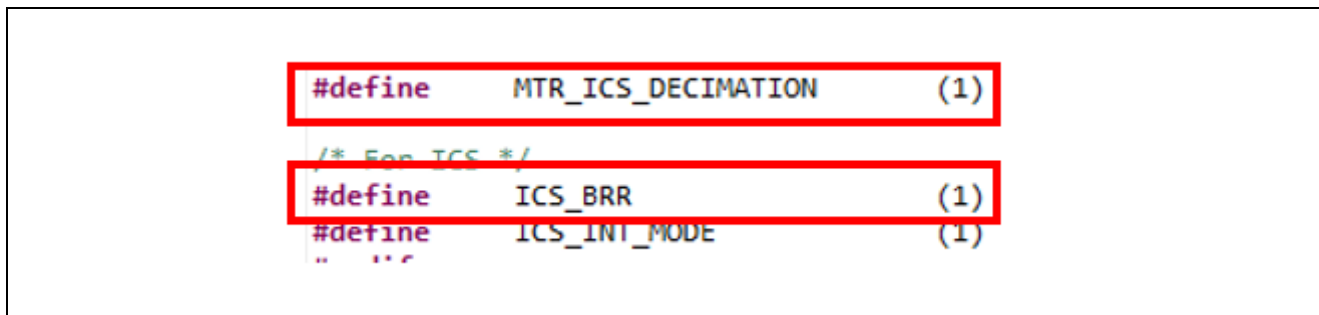

Figure 4-5 Modification of r_mtr_ics.h

- Change the communication speed setting of Renesas Motor Workbench to connect

(1) Press the Clock button on the Main Window to change the value to 80,000,000
   This value was calculated by multiplying the default 8,000,000 by 10
   because the UART communication baud rate was changed from 1Mbps to 10Mbps.

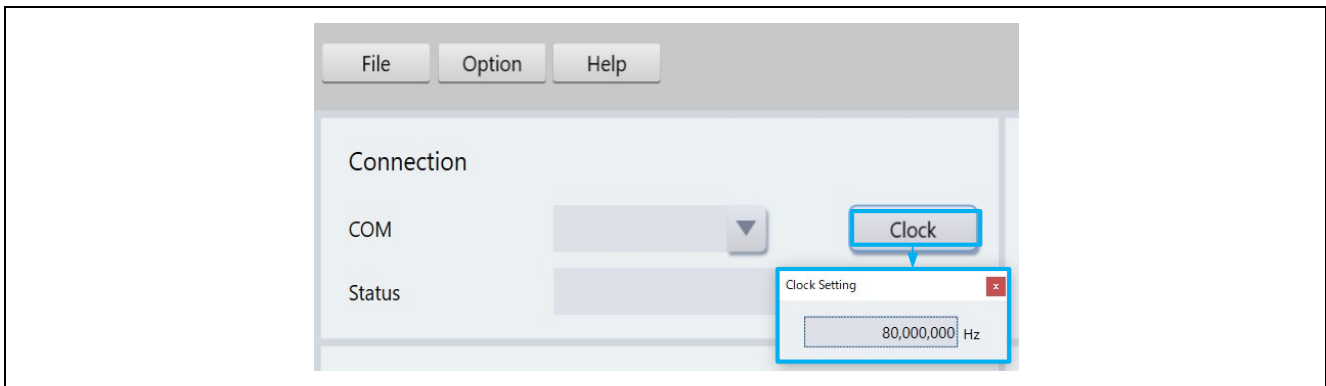(2) Select the COM of the connected kit in the COM of Connection

Figure 4-6 Clock frequency setting

If the connection fails, repeat the procedure for reconnecting after resetting the communication board.

5. Reference Documents

- RA6T1 Group User's Manual: Hardware (R01UH0897)
- Renesas Flexible Software Package User's manual
  (PDF version : R11UM0155, Web version : RA Flexible Software Package Documentation)
- Renesas e2 studio 2022-07 or higher User's Manual: Quick Start Guide (R20UT5210)
- Application note: Sensorless vector control for permanent magnet synchronous motor (Algorithm)
  (R01AN3786)
- Renesas Motor Workbench User's Manual (R21UZ0004)
- Renesas Motor Workbench Quick start guide (R21QS0011)
- Motor Control Evaluation System for RA Family (R12UZ0078)
- RA6T1 CPU Card User's Manual (R12UZ0077)

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Aug.30, 2021	-	First edition issued
1.01	Oct. 18, 2022	-	For FSP V4.1.0
1.10	Aug.30, 2023	-	• Updated for Evaluation System for BLDC Motor Series • Support for Dual Motor in FSP V4.6.0 rm_motor_driver module

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

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

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.

3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.

4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.

5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.

6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).

7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.

8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.

2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.

3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.

4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.

5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.

6. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.
   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.

7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION ("Vulnerability Issues"). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.

8. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.

9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.

10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.

11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.

12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.

13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.

14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

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

www.renesas.com

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/.