R01AN4720EJ0100
Rev.1.00
Mar. 29, 2019

RX72T

Sensorless Vector Control for Permanent Magnet Synchronous Motor (Implementation)

Abstract

This application note aims at explaining sensorless vector control software for a permanent magnet synchronous motor, by using functions of RX72T. The explanation includes, how to use the library of ‘Renesas Motor Workbench’ tool, that is support tool for motor control development.

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

Operation Checking Device

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

- RX72T (R5F572TKCDFB)

Target Software

The target software of this application note is as follows.

- RX72T_MRSSK_SPM_LESS_FOC_CSP_RV100 (IDE：CS+)

- RX72T_MRSSK_SPM_LESS_FOC_E2S_RV100 (IDE：e^{2}studio)

RX72T Sensorless vector control software for ‘24V Motor Control Evaluation System for RX23T’ and ‘RX72T CPU Card’

Reference

- RX72T Group User’s Manual: Hardware (R01UH0803)

- Application note: ‘Sensorless vector control for permanent magnet synchronous motor (Algorithm)’ (R01AN3786)

- Renesas Motor Workbench User’s Manual (R21UZ0004)

- Renesas Solution Starter Kit 24V Motor Control Evaluation System for RX23T User’s Manual (R20UT3697)

Contents

1. Overview

This application note explains how to implement the sensorless vector control software of permanent magnet synchronous motor (PMSM) using the RX72T microcontroller. The explanation includes, how to use the library of 'Renesas Motor Workbench' tool, that is support tool for motor control development.

Note that the software uses the algorithm described in the application note 'Sensorless vector control for permanent magnet synchronous motor (Algorithm)'.

1.1 Development Environment

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

Table 1-1 Hardware Development Environment

Microcontroller	Evaluation board	Motor
RX72T (R5F572TKCDFB)	24V inverter board & RX72T CPU Card (Note 1)	TG-55L (Note 2)

Table 1-2 Software Development Environment

Toolchain version
CC-RX: V3.01.00

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

Notes:1. 24V inverter board is products of Renesas Electronics Corporation.
2. TG-55L is the product of TSUKASA ELECTRIC.
TSUKASA ELECTRIC (http://www.tsukasa-d.co.jp/)

2. System Overview

Overview of this system is explained below.

2.1 Hardware Configuration

The hardware configuration is shown below.

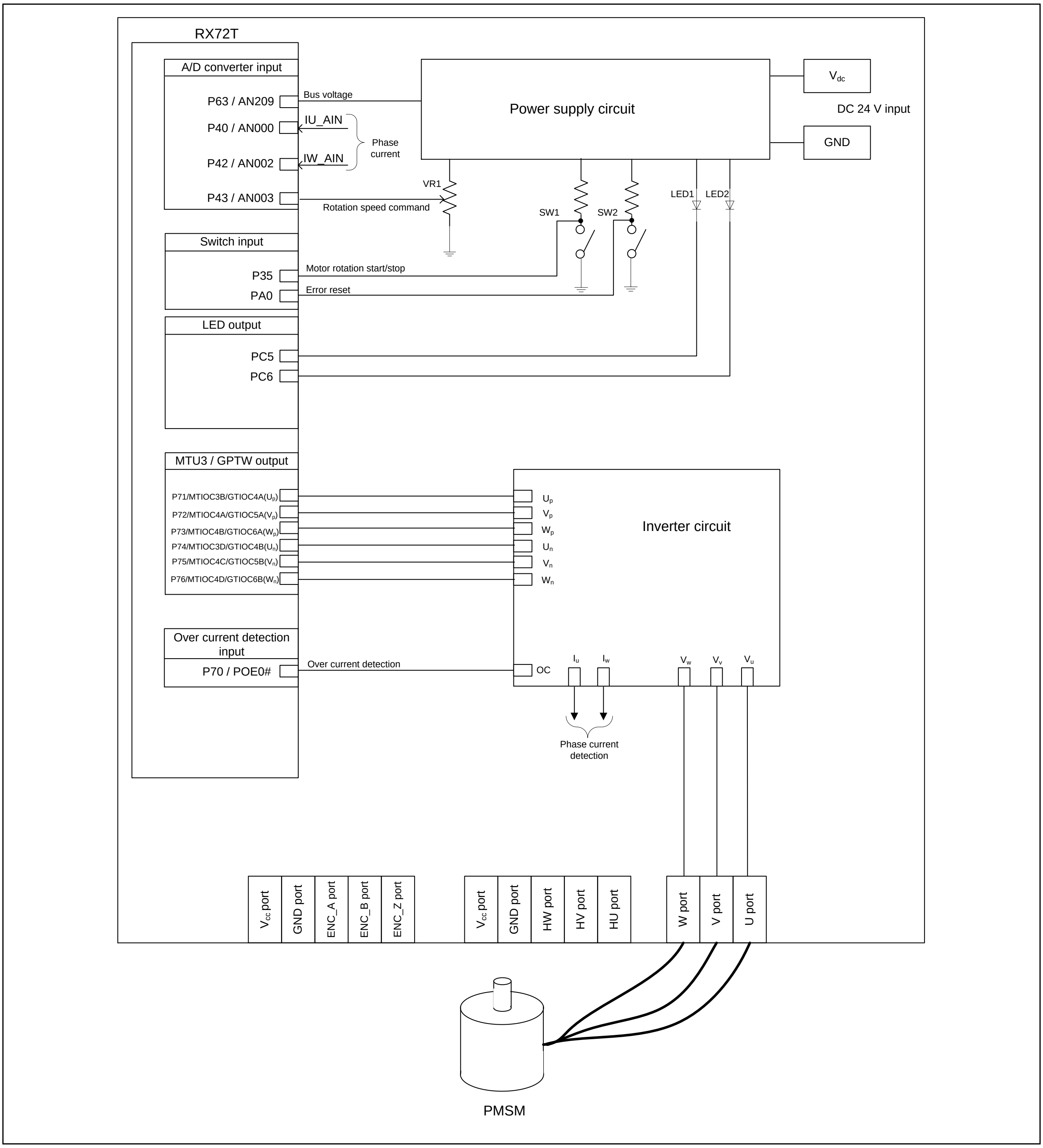


Figure 2-1 Hardware Configuration Diagram

2.2 Hardware Specifications

2.2.1 User Interface

List of user interfaces of this system is given in Table 2-1.

Table 2-1 User Interfaces

Item	Interface component	Function
Rotation speed	Variable resistor (VR1)	Reference value of rotation speed input (analog value)
START/STOP	Toggle switch (SW1)	Motor rotation start/stop command
ERROR RESET	Toggle switch (SW2)	Command of recovery from error status
LED1	Yellow green LED	- At the time of motor rotation: ON - At the time of stop: OFF
LED2	Yellow green LED	- At the time of error detection: ON - At the time of normal operation: OFF
RESET	Push switch	System reset

List of port interfaces of this system is given in Table 2-2.

Table 2-2 Port Interfaces

R5F572TKCDFB port name	Function
P63 / AN209	Inverter bus voltage measurement
P43 / AN003	For rotation speed command value input (analog value)
P35	START/STOP toggle switch
PA0	ERROR RESET toggle switch
PC5	LED1 ON/OFF control
PC6	LED2 ON/OFF control
P40 / AN000	U phase current measurement
P42 / AN002	W phase current measurement
P71 / MTIOC3B / GTIOC4A	PWM output (U_p)
P72 / MTIOC4A / GTIOC5A	PWM output (V_p)
P73 / MTIOC4B / GTIOC6A	PWM output (W_p)
P74 / MTIOC3D / GTIOC4B	PWM output (U_n)
P75 / MTIOC4C / GTIOC5B	PWM output (V_n)
P76 / MTIOC4D / GTIOC6B	PWM output (W_n)
P70 / POE0#	PWM emergency stop input at the time of over-current detection

2.2.2 Peripheral Functions

List of the peripheral functions used in this system is given in Table 2-3.

Table 2-3 List of the Peripheral Functions

12-bit A/D Converter	CMT	MTU3 / GPTW	POE3B
- Rotation speed command value input - Current of each phase U and W measurement - Inverter bus voltage measurement	500 [μs] interval timer	Complementary PWM output	Set PWM output ports to high impedance state to stop the PWM output.

(1) 12-Bit A/D Converter (S12ADH)

U phase current (Iu), W phase current (Iw), inverter bus voltage (Vdc) and rotation speed reference are measured by using the single scan mode (use hardware trigger). The sample-and-hold function is used for U phase current (Iu) and W phase current (Iw) measurement.

(2) Compare Match Timer (CMT)

The channel 0 of the compare match timer is used as 500 [μs] interval timer.

(3) Multi-Function Timer Pulse Unit 3 (MTU3d)

On the channel 3 and 4, output (active level: high) with dead time is performed by using the complementary PWM mode.

(4) General PWM Timer (GPTW)

On the channel 4, 5 and 6, output (active level: high) with dead time is performed by using the complementary PWM Output Operating Mode.

(5) Port Output Enable 3 (POE3B)

PWM output ports are set to high impedance state when an over-current is detected (when a falling edge of the POE0# port is detected) or when an output short circuit is detected.

The setting of the PWM output timer is selected by the following macro definition.

Table 2-4 List of Macro Definitions 'r_mtr_ctrl_rx72t.h'

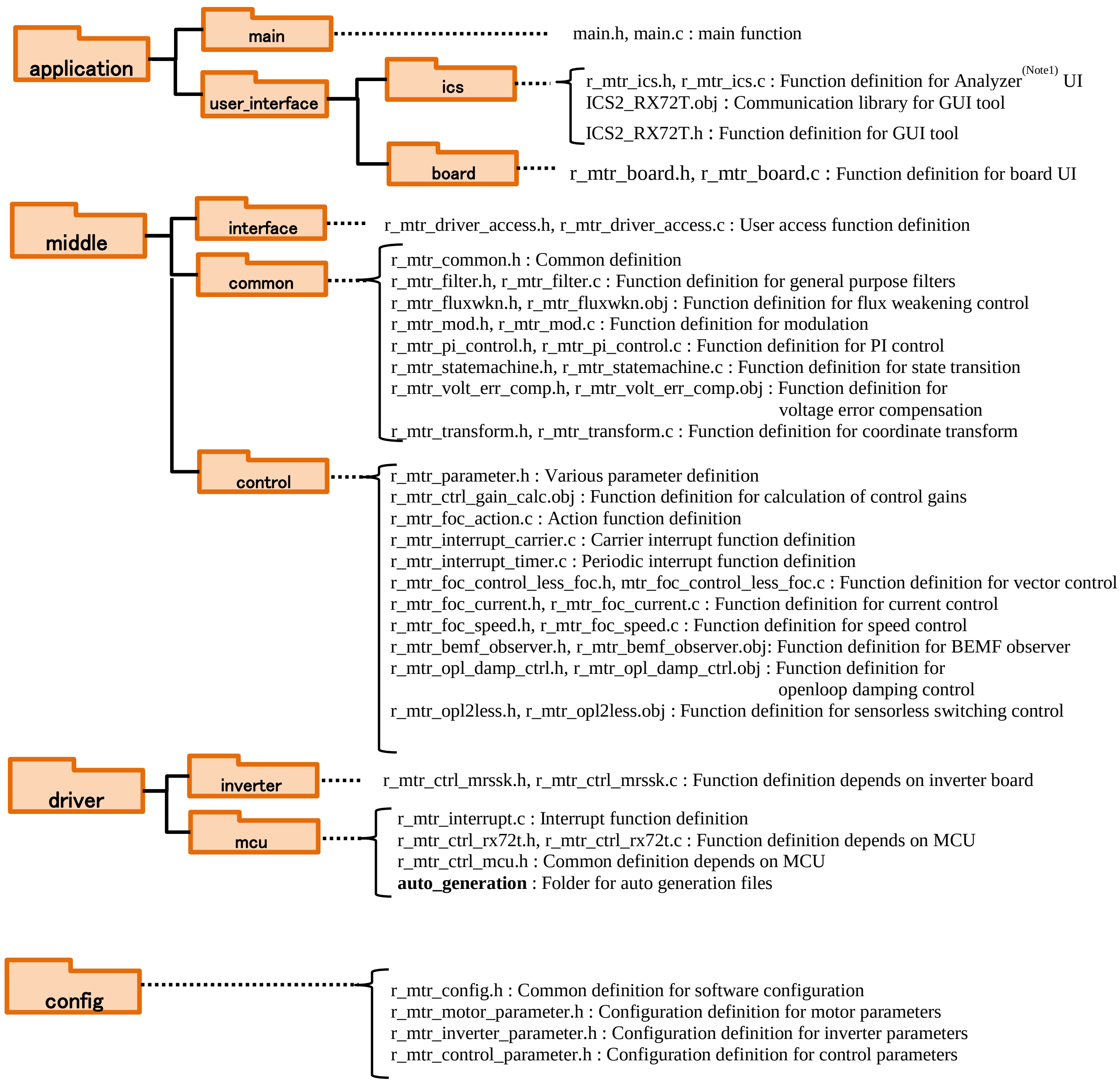
File name	Macro name	Definition value	Remarks
r_mtr_ctrl_rx72t.h	MTR_GPT	0	0:MTU 1:GPT

2.3 Software Configuration

2.3.1 Software File Configuration

Folder and file configuration of the software are given below.

(Project Folder)



Note 1: Regarding the specification of Analyzer function in the motor control development support tool ‘Renesas Motor Workbench’, please refer to the chapter 4.The identifier ‘ics/ICS (ICS is previous motor control development support tool ‘In Circuit Scope’) is attached to the name of folders, files, functions, variables related to ‘Renesas Motor Workbench’.

Figure 2-2 Folder and File Configuration

2.3.2 Module Configuration

Module configuration of the software is described below.

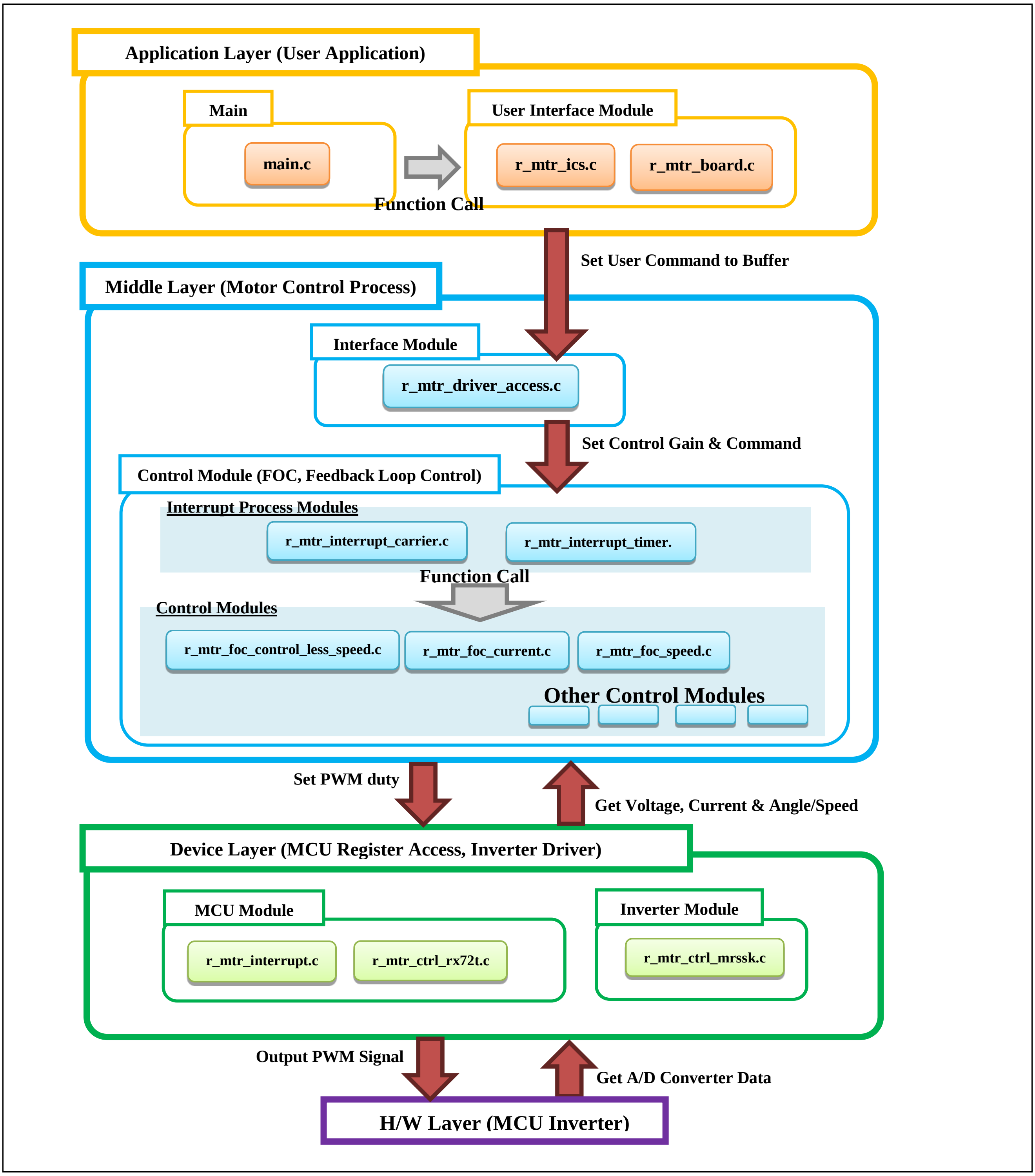


Figure 2-3 Module Configuration

2.4 Software Specifications

Table 2-5 shows basic software specification of this system. For details of the sensorless vector control, refer to the application note ‘Sensorless vector control for permanent magnet synchronous motor (Algorithm)’.

Table 2-5 Basic Specifications of Sensorless Vector Control Software

<table>
<tr><th>Item</th><th colspan="2">Content</th></tr>
<tr><td>Control method</td><td colspan="2">Vector control</td></tr>
<tr><td>Position detection method</td><td colspan="2">Sensorless</td></tr>
<tr><td>Motor rotation start/stop</td><td colspan="2">Determined depending on the level of SW1 (‘Low’: rotation start, ‘High’: stop) or input from Analyzer</td></tr>
<tr><td>Input voltage</td><td colspan="2">DC 24 [V]</td></tr>
<tr><td>Carrier frequency (PWM)</td><td colspan="2">20 [kHz]</td></tr>
<tr><td>Dead time</td><td colspan="2">2 [μs]</td></tr>
<tr><td>Control period</td><td colspan="2">Current control / Position and speed estimation: 50 [μs]
Speed control:500 [μs]</td></tr>
<tr><td>Rotation speed control range</td><td colspan="2">CW: 0 [rpm] to 2650 [rpm]
CCW: 0 [rpm] to 2650 [rpm]</td></tr>
<tr><td>Natural frequency of each control system</td><td colspan="2">Current control system: 300 [Hz]
Speed control system: 5 [Hz]
BEMF estimation system: 1000 [Hz]
Position estimation system: 50 [Hz]</td></tr>
<tr><td rowspan="2">Optimization setting of compiler</td><td>Optimization level</td><td>2(-optimize=2) (default setting)</td></tr>
<tr><td>Optimization method</td><td>Size priority(-size) (default setting)</td></tr>
<tr><td>ROM/RAM size</td><td colspan="2">ROM: 17.3KB
RAM: 4.5KB</td></tr>
<tr><td>Processing stop for protection</td><td colspan="2">- Disables the motor control signal output (six outputs), under any of the following conditions.
1. Current of each phase exceeds 0.89 [A] (monitored every 50 [μs])
2. Inverter bus voltage exceeds 28 [V] (monitored every 50 [μs])
3. Inverter bus voltage is less than 14 [V] (monitored every 50 [μs])
4. Rotation speed exceeds 3000 [rpm] (monitored every 50 [μs])

- When an external over-current signal is detected (when a falling edge of the POE0# port is detected) or when the output short circuit is detected, the PWM output ports are set to high impedance state.</td></tr>
</table>

3. Descriptions of the Control Program

The target software of this application note is explained here.

3.1 Contents of Control

3.1.1 Motor Start/Stop

The start and stop of the motor are controlled by input from Analyzer function of 'Renesas Motor Workbench' or SW1 switch of RSSK board.

A general-purpose port is assigned to SW1. The port is read within the main loop. When the port is at a 'Low' level, the software determines that the motor should be started. Conversely, when the level is switched to 'High' level, the software determines that the motor should be stopped.

3.1.2 A/D Converter

(1) Motor Rotation Speed Reference

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

Table 3-1 Conversion Ratio of the Rotation Speed Reference

Item	Conversion ratio (Reference: A/D conversion value)		Channel
Rotation speed reference	CW	0 rpm to 2700 rpm: 0800H to 0FFFH	AN003
	CCW	0 rpm to 2700 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-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)	Channel
Inverter bus voltage	0 [V] to 111 [V]: 0000H to 0FFFH	AN209

(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 current: A/D conversion value)	Channel
U, W phase current	-10 [A] to 10 [A]: 0000H to 0FFFH (Note)	Iu: AN000 Iw: AN002

Note: For more details of A/D conversion characteristics, refer to 'RX72T Group User's Manual: Hardware'.

3.1.3 Modulation

The target software of this application note uses pulse width modulation (hereinafter called PWM) to generate the input voltage to the motor. And the PWM waveform is generated by the triangular wave comparison method.

(1) Triangular Wave Comparison Method

The triangular wave comparison method is used to output the voltage command value. By this method, the pulse width of the output voltage can be determined by comparing the carrier waveform (triangular wave) and voltage command value waveform. The voltage command value of the pseudo sinusoidal wave can be output by turning the switch on or off when the voltage command value is larger or smaller than the carrier wave respectively.

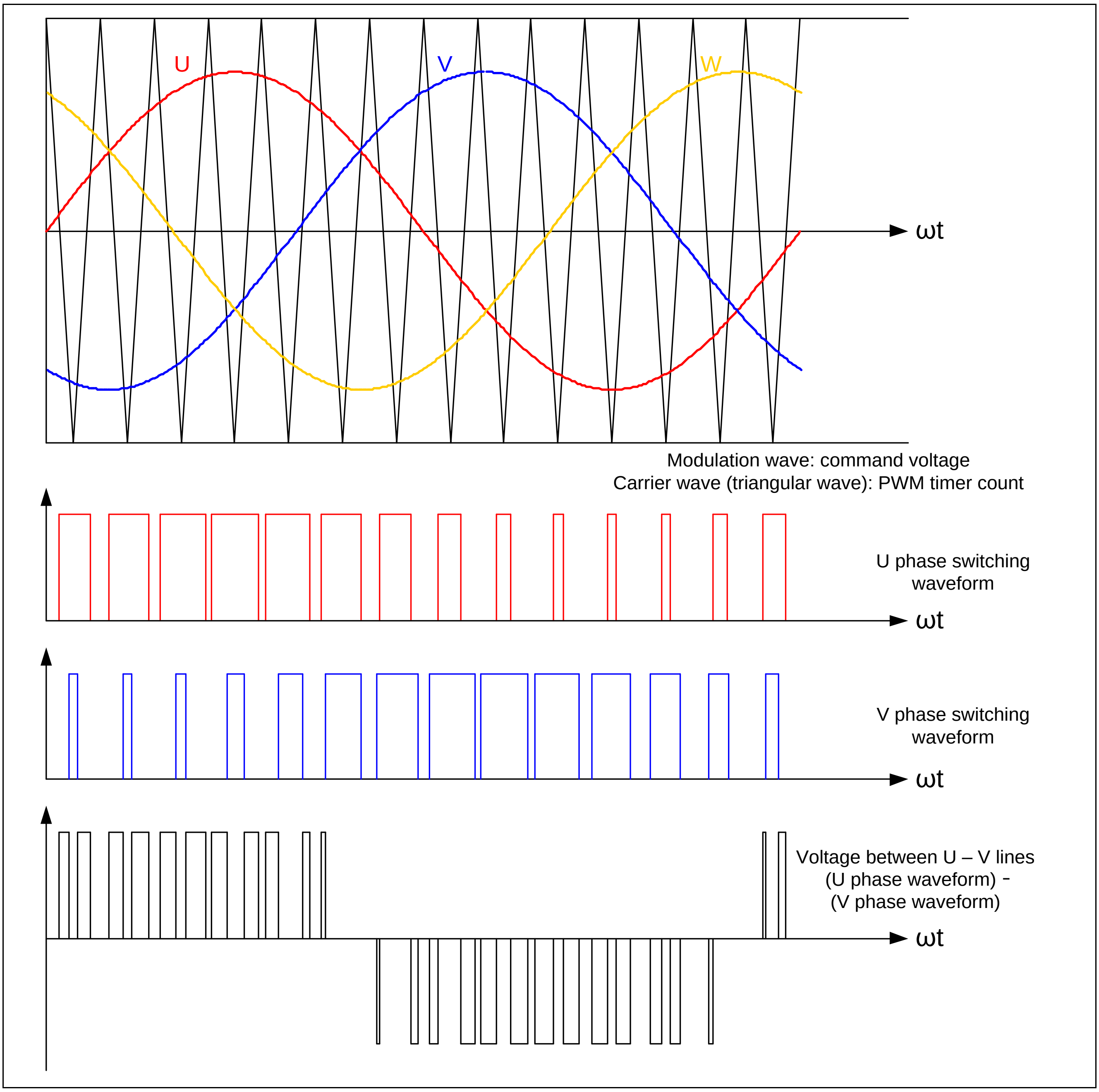


Figure 3-1 Conceptual Diagram of the Triangular Wave Comparison Method

As shown in Figure 3-2, ratio of the output voltage pulse to the carrier wave cycle is called duty.

$$\text{Duty} = \frac{T_{ON}}{T_{ON} + T_{OFF}} \times 100\ [\%]$$

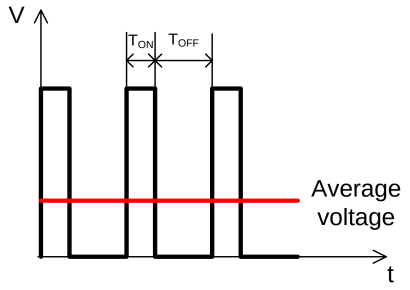


Figure 3-2 Definition of Duty

Modulation factor m is defined as follows.

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

m: Modulation factor V: Voltage command value E: Inverter bus voltage

The voltage command can be generated by setting PWM compare register properly to obtain the desired duty.

3.1.4 State Transition

Figure 3-3 is a state transition diagram of the sensorless vector control software. In the target software of this application note, the software state is managed by 'SYSTEM MODE' and 'RUN MODE'. And 'Control Config' shows the active control system in the software.

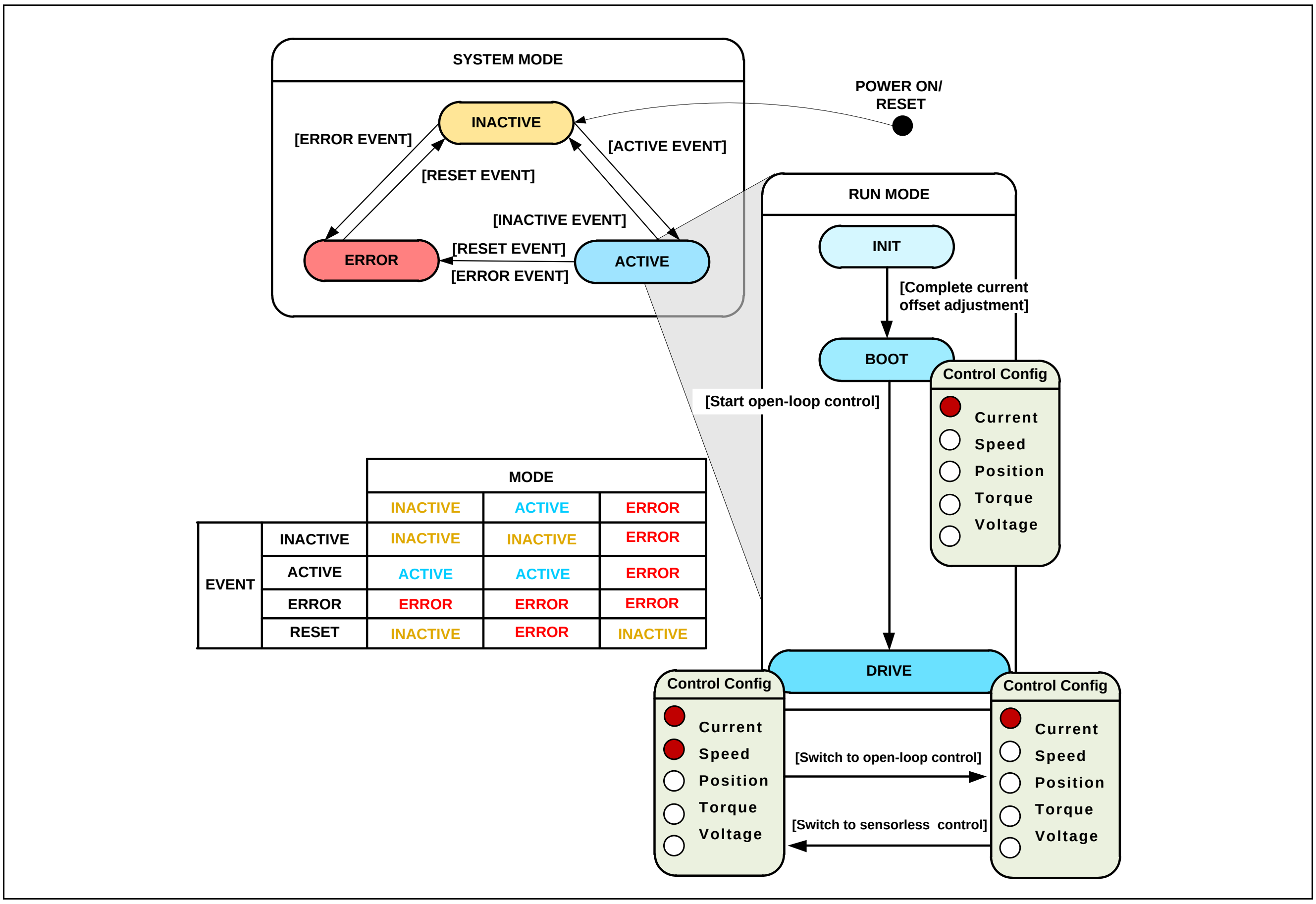


Figure 3-3 State Transition Diagram of Sensorless Vector Control Software

(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). RUN MODE

'RUN MODE' indicates the condition of the motor control. 'RUN MODE' transits sequentially as shown in Figure 3-3 when 'SYSTEM MODE' is 'ACTIVE'.

(3). EVENT

When 'EVENT' occurs in each 'SYSTEM MODE', 'SYSTEM MODE' changes as shown in the table of Figure 3-3, according to that 'EVENT'.

Table 3-4 List of EVENT

EVENT name	Occurrence factor
INACTIVE	By user operation
ACTIVE	By user operation
ERROR	When the system detects an error
RESET	By user operation

3.1.5 Startup Method

Figure 3-4 shows startup control of sensorless vector control software. Each reference value setting of d-axis current, q-axis current and speed is managed by respective status.

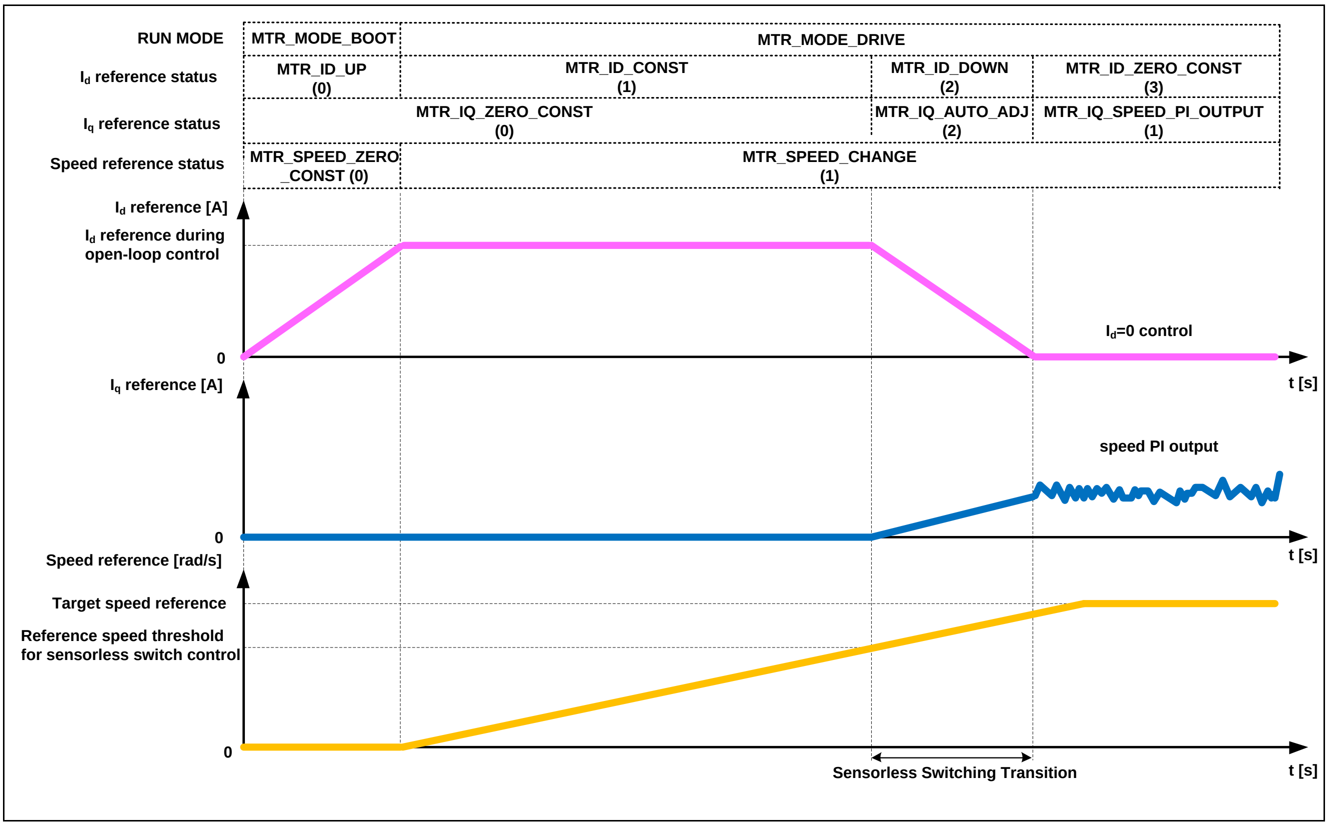


Figure 3-4 Startup Control of Sensorless Vector Control Software

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 setting value for the system protection function.

- Over-current error

The over-current detection is performed by both hardware detection method and software detection method. In response to over-current detection, an emergency stop signal is generated from the hardware (hardware detection). When the emergency stop signal is generated, the PWM output ports are set to high impedance state.

In addition, U, V, and W phase currents are monitored in every over-current monitoring cycle. When an over-current is detected, the CPU executes emergency stop (software detection). The over-current limit value is calculated from the nominal current of the motor [MP_NOMINAL_CURRENT_RMS].

- Over-voltage error

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

- Under-voltage error

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

- Over-speed error

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

Table 3-5 Setting Values of the System Protection Function

Over-current error	Over-current limit value [A]	0.89
	Monitoring cycle [μs]	50
Over-voltage error	Over-voltage limit value [V]	28
	Monitoring cycle [μs]	50
Under-voltage error	Under-voltage limit value [V]	14
	Monitoring cycle [μs]	50
Over-speed error	Speed limit value [rpm]	3000
	Monitoring cycle [μs]	50

3.2 Function Specifications of Sensorless Vector Control Software

The control process of the target software of this application note is mainly consisted of 50[μs] period interrupt (carrier interrupt) and 500[μs] period interrupt. In Figure 3-5 and Figure 3-6, the control process in the red broken line part is executed every 50[μs] period, and the control process in the blue broken line part is executed every 500[μs] period.

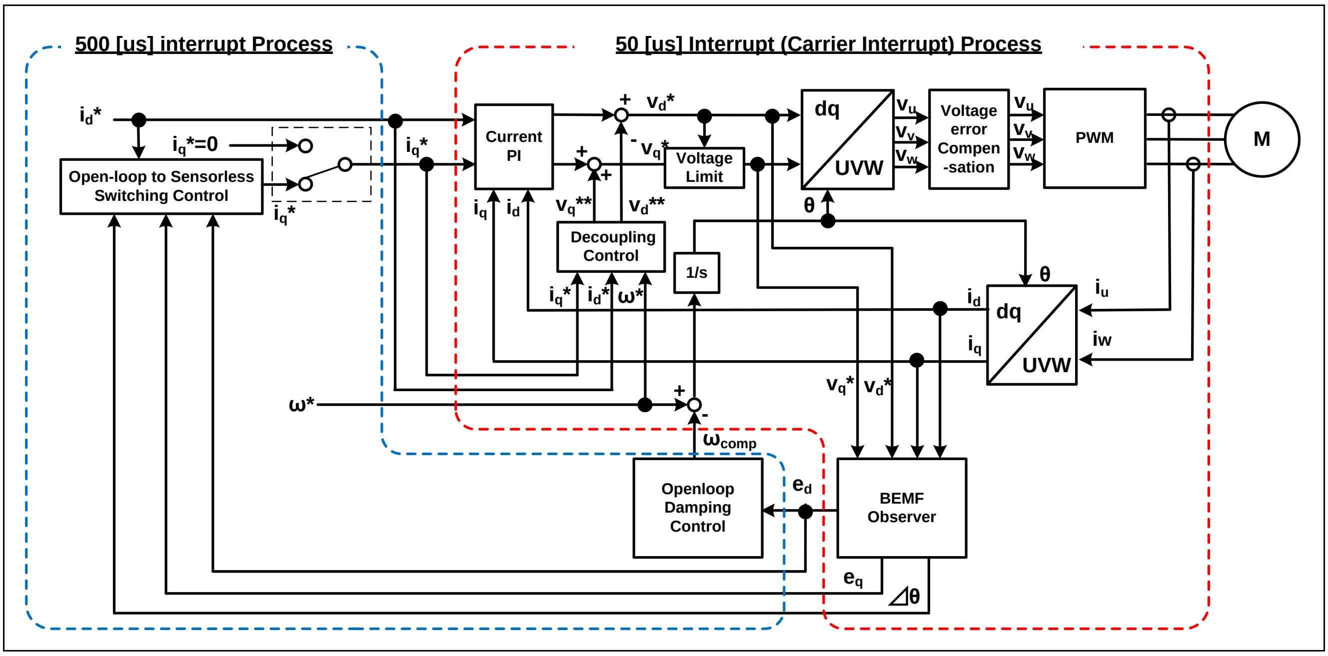


Figure 3-5 Block Diagram of Sensorless Vector Control (Open-loop Control)

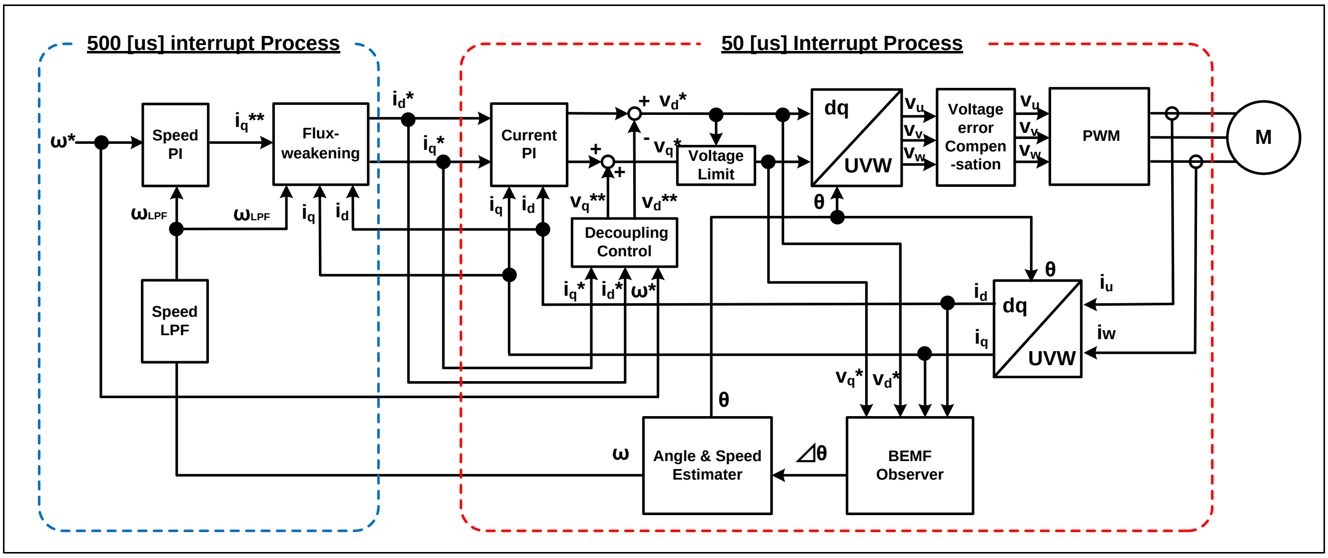


Figure 3-6 Block Diagram of Sensorless Vector Control (Sensorless Control)

This chapter shows the specification of 2 interrupt functions and functions executed in each interrupt period. In the following tables, only primary functions of the sensorless vector control are listed. Regarding the specification of functions not listed in the following tables, refer to source codes.

Table 3-6 List of Interrupt Functions

File name	Function overview	Process overview
r_mtr_interrupt_carrier.c	mtr _foc_interrupt_carrier Input : (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output : None	Calling every 50 [μs] - Current and Vdc monitoring - Current PI control - Speed/position estimation - PWM duty setting
r_mtr_interrupt_timer.c	mtr_foc_interrupt_500us Input : (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output : None	Calling every 500 [μs] - Startup control - d-axis/q-axis current and speed reference setting - Speed PI control

Table 3-7 List of Functions Executed in 50[μs] Period Interrupt (Carrier Interrupt) (1/2)

File name	Function overview	Process overview
r_mtr_ctrl_mrssk.c	mtr_get_current_iuiw Input: (float*) f4_iu_ad / U phase current A/D conversion value pointer (float*) f4_iw_ad / W phase current A/D conversion value pointer (uint8_t) u1_id / Motor ID Output: None	Obtaining the U/W phase current
	mtr_get_vdc Input: (uint8_t) u1_id / Motor ID Output: (float) f4_temp_vdc / Vdc value	Obtaining the Vdc
r_mtr_foc_control_less_speed.c	mtr_error_check Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: None	Error monitoring
	mtr_current_offset_adjustment Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: None	Cancel current offset
	mtr_calib_current_offset Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: None	Calculation of current offset
	mtr_angle_speed Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: None	Position and speed estimation
	mtr_foc_voltage_limit Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: None	Limiting voltage reference
r_mtr_foc_current.c	mtr_current_pi_control Input: (mtr_current_control_t *) st_cc / Structure pointer for current control Output: None	Current PI control
	mtr_foc_current_decoupling Input: (mtr_current_control_t *) st_cc / Structure pointer for current control (float)f4_speed_rad / Rotation speed (const mtr_parameter_t *) p_mtr / Structure pointer for motor parameter Output: None	Decoupling control
r_mtr_transform.c	mtr_transform_uvw_dq_abs Input: (const mtr_rotor_angle_t *) p_angle / Structure pointer for phase management (const float*) f4_uvw / UVV phase pointer (float*) f4_dq / dq-axis pointer Output: None	Coordinate transform UVW to dq
	mtr_transform_dq_uvw_abs Input: (const mtr_rotor_angle_t *) p_angle / Structure pointer for phase management (const float*) f4_dq / dq-axis pointer (float*) f4_uvw / UVW phase pointer Output: None	Coordinate transform dq to UVW

Table 3-8 List of Functions Executed in 50[μs] Period Interrupt (Carrier Interrupt) (2/2)

File name	Function overview	Process overview
r_mtr_volt_err_comp.obj	mtr_volt_err_comp_main Input: (mtr_volt_comp_t *) st_volt_comp / Structure pointer for voltage error compensation (float*) p_f4_v_array / Array pointer for 3-phase voltage compensation amount (float*) p_f4_i_array / Array pointer for 3 phase current (float) f4_vdc / Inverter bus voltage Output: None	Voltage error compensation
r_mtr_ctrl_rx72t.c	mtr_inv_set_uvw Input: (float) f4_duty_u / U phase modulation factor (float) f4_duty_v / V phase modulation factor (float) f4_duty_w / W phase modulation factor (uint8_t) u1_id / Motor ID Output: None	PWM output setting
r_mtr_bemf_observer.obj	mtr_bemf_observer Input: (mtr_bemf_observer_t *) st_bemf_obs / Structure pointer for BEMF observer (float) f4_vd_ref / d-axis voltage reference (float) f4_vq_ref / q-axis voltage reference (float) f4_id / d-axis current (float) f4_iq / q-axis current Output: None	Calculation for BEMF observer
	mtr_bemf_calc_d Input: (mtr_bemf_observer_t *) st_bemf_obs / Structure pointer for BEMF observer (float) f4_speed_rad / Estimated speed (float) f4_iq / q-axis current Output: (float) f4_temp / Estimated d-axis BEMF	Calculation for estimated d-axis BEMF
	mtr_bemf_calc_q Input: (mtr_bemf_observer_t *) st_bemf_obs / Structure pointer for BEMF observer (float) f4_speed_rad / Estimated speed (float) f4_id / d-axis current Output: (float) f4_temp / Estimated q-axis BEMF	Calculation for estimated q-axis BEMF
	mtr_angle_speed_pll Input: (mtr_pll_est_t *) st_pll_est / Structure pointer for position and speed estimation (float)f4_phase_err / Phase error (float*) f4_speed / Estimated speed pointer Output: None	Calculation for position and speed estimation
r_mtr_opl_damp_ctrl.obj	mtr_opl_damp_ctrl Input: (mtr_opl_damp_t *) st_opl_damp / Structure pointer for open-loop damping control (float) f4_ed / Estimated d-axis BEMF (float) speed_ref / Speed reference Output: (float)f4_temp_damp_comp_speed / Feedback value for speed reference	Open-loop damping control

Table 3-9 List of Functions Executed in 500[μs] Period Interrupt

File name	Function overview	Process overview
r_mtr_foc_control_less_speed.c	mtr_set_speed_ref Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: (float) f4_speed_rad_ref_buff / Speed reference	Speed reference setting
	mtr_set_iq_ref Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: (float) f4_iq_ref_buff / q-axis current reference	q-axis current reference setting
	mtr_set_id_ref Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: (float) f4_id_ref_buff / d-axis current reference	d-axis current reference setting
r_mtr_foc_speed.c	mtr_speed_pi_control Input: (mtr_speed_control_t *) st_sc / Structure pointer for speed control (float) f4_speed_rad / Rotation speed Output: (float) f4_iq_ref_calc / q-axis current reference	Speed PI control
r_mtr_opl2less.obj	mtr_opl2less_iq_calc input: (float) f4_ed / Estimated d-axis BEMF (float) f4_eq / Estimated q-axis BEMF (float) f4_id / d-axis current reference when open-loop (float) f4_torque_current / Torque current when open-loop control (float) f4_phase_err / Phase error Output: (float) f4_temp_iq_ref / q-axis current reference	Generating q-axis current reference for sensorless switching control
r_mtr_fluxwkn.obj	R_FLUXWKN_Run Input: (fluxwkn_t *) p_fluxwkn / Structure pointer for flux weakening control (float) f4_speed_rad / Rotation speed (const float*) p_f4_idq / dq-axis current pointer (float*) p_f4_idq_ref / dq-axis current reference pointer Output: (uint16_t) u2_fw_status / Status of flux-weakening control	Flux-weakening control

3.3 Macro Definition of Sensorless Vector Control Software

The macro definitions in the target software of this application note are listed below. In the following tables, only definitions set the software configuration are listed. Regarding the macro definitions not listed in the following tables, refer to source codes.

Table 3-10 List of Macro Definitions ‘r_mtr_motor_parameter.h’

File name	Macro name	Definition value	Remarks
r_motor_parameter.h	MP_POLE_PAIRS	2	Number of pole pairs
	MP_MAGNETIC_FLUX	0.02159f	Flux [Wb]
	MP_RESISTANCE	8.5f	Resistance [Ω]
	MP_D_INDUCTANCE	0.0045f	d-axis inductance [H]
	MP_Q_INDUCTANCE	0.0045f	q-axis inductance [H]
	MP_ROTOR_INERTIA	0.0000028f	Rotor inertia [kgm^2]
	MP_NOMINAL_CURRENT_RMS	0.42f	Nominal current [A(rms)]

Table 3-11 List of Macro Definitions ‘r_mtr_control_parameter.h’

File name	Macro name	Definition value	Remarks
r_mtr_control_parameter.h	CP_CURRENT_OMEGA	300.0f	Natural frequency of current control system [Hz]
	CP_CURRENT_ZETA	1.0f	Damping ratio of current control system
	CP_SPEED_OMEGA	5.0f	Natural frequency of speed control system [Hz]
	CP_SPEED_ZETA	1.0f	Damping ratio of speed control system
	CP_E_OBS_OMEGA	1000.0f	Natural frequency of BEMF estimation system [Hz]
	CP_E_OBS_ZETA	1.0f	Damping ratio of BEMF estimation system
	CP_PLL_EST_OMEGA	20.0f	Natural frequency of position estimation system [Hz]
	CP_PLL_EST_ZETA	1.0f	Damping ratio of position estimation system
	CP_ID_DOWN_SPEED_RPM	600	Speed (mechanical) when start decreasing d-axis current reference [rpm]
	CP_ID_UP_SPEED_RPM	500	Speed (mechanical) when start increasing d-axis current reference [rpm]
	CP_MAX_SPEED_RPM	2650	Maximum speed (mechanical) [rpm]
	CP_OVERSPEED_LIMIT_RPM	3000	Speed limit value (mechanical) [rpm]
	CP_OL_ID_REF	0.3f	d-axis current reference in open-loop mode [A]

Table 3-12 List of Macro Definitions 'r_mtr_inverter_parameter.h'

File name	Macro name	Definition value	Remarks
r_mtr_inverter_parameter.h	IP_DEADTIME	2.0f	Dead time [μs]
	IP_CURRENT_RANGE	20.0f	Current A/D conversion range [A]
	IP_VDC_RANGE	111.0f	Vdc A/D conversion range [V]
	IP_INPUT_V	24.0f	Input DC voltage [V]
	IP_CURRENT_LIMIT	5.0f	Over-current limit [A](Note)
	IP_OVERVOLTAGE_LIMIT	28.0f	High voltage limit [V]
	IP_UNDERVOLTAGE_LIMIT	14.0f	Low voltage limit [V]

Note: This value is calculated from the rated power of the shunt resistance.

Table 3-13 List of Macro Definitions 'r_mtr_config.h'

File name	Macro name	Definition value	Remarks
r_mtr_config.h	RX72T_MRSSK	-	MCU select macro
	IP_MRSSK	-	Inverter select macro
	MP_TG55L	-	Motor select macro
	CP_TG55L	-	
	CONFIG_DEFAULT_UI	ICS_UI	Default UI selection ICS_UI: Use Analyzer UI BOARD_UI: Board UI
	FUNC_ON	1	Enable
	FUNC_OFF	0	Disable
	DEFAULT_LESS_SWITCH	FUNC_ON	Sensorless switching control
	DEFAULT_FLUX_WEAKENING	FUNC_OFF	Flux weakening control
	DEFAULT_VOLT_ERR_COMP	FUNC_ON	Voltage error compensation
	DEFAULT_OPENLOOP_DAMPING	FUNC_ON	Open-loop damping control
	GAIN_MODE	MTR_GAIN_DESIGN_MODE	Gain mode MTR_GAIN_DESIGN_MOD: PI gain design mode MTR_GAIN_DIRECT_MODE: PI gain direct input mode
	MOD_METHOD	MOD_METHOD_SVPWM	Modulation method MOD_METHOD_SPWM: Sinusoidal PWM MOD_METHOD_SVPWM: Space Vector PWM

Table 3-14 List of Macro Definitions 'r_mtr_common.h'

File name	Macro name	Definition value	Remarks
r_mtr_common.h	MTR_TFU_OPTIMIZE	1	1:Use TFU code 0:Use Standard library code

3.4 Control Flowcharts

3.4.1 Main Process

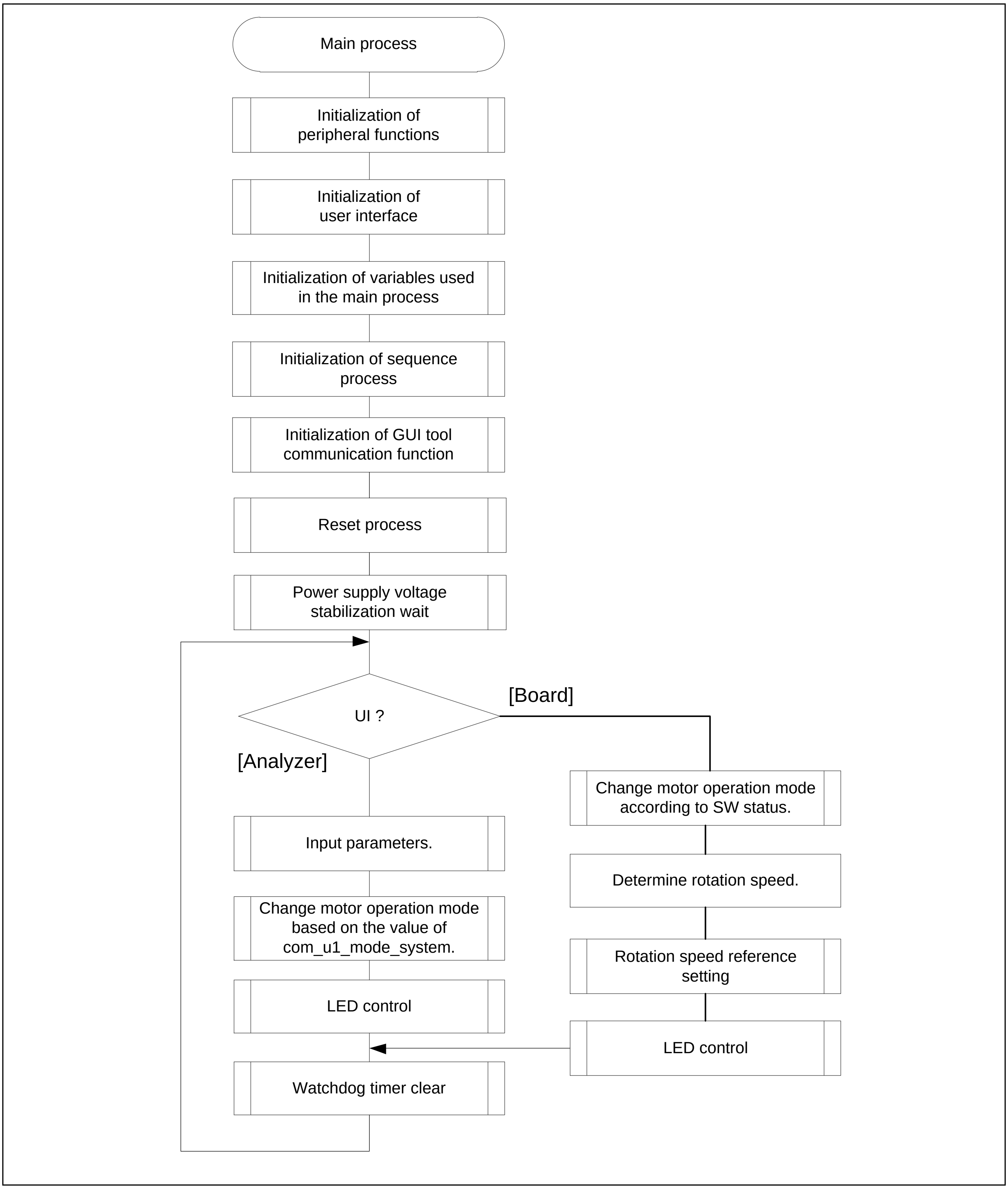


Figure 3-7 Main Process Flowchart

3.4.2 50[μs] Period Interrupt (Carrier Interrupt) Process

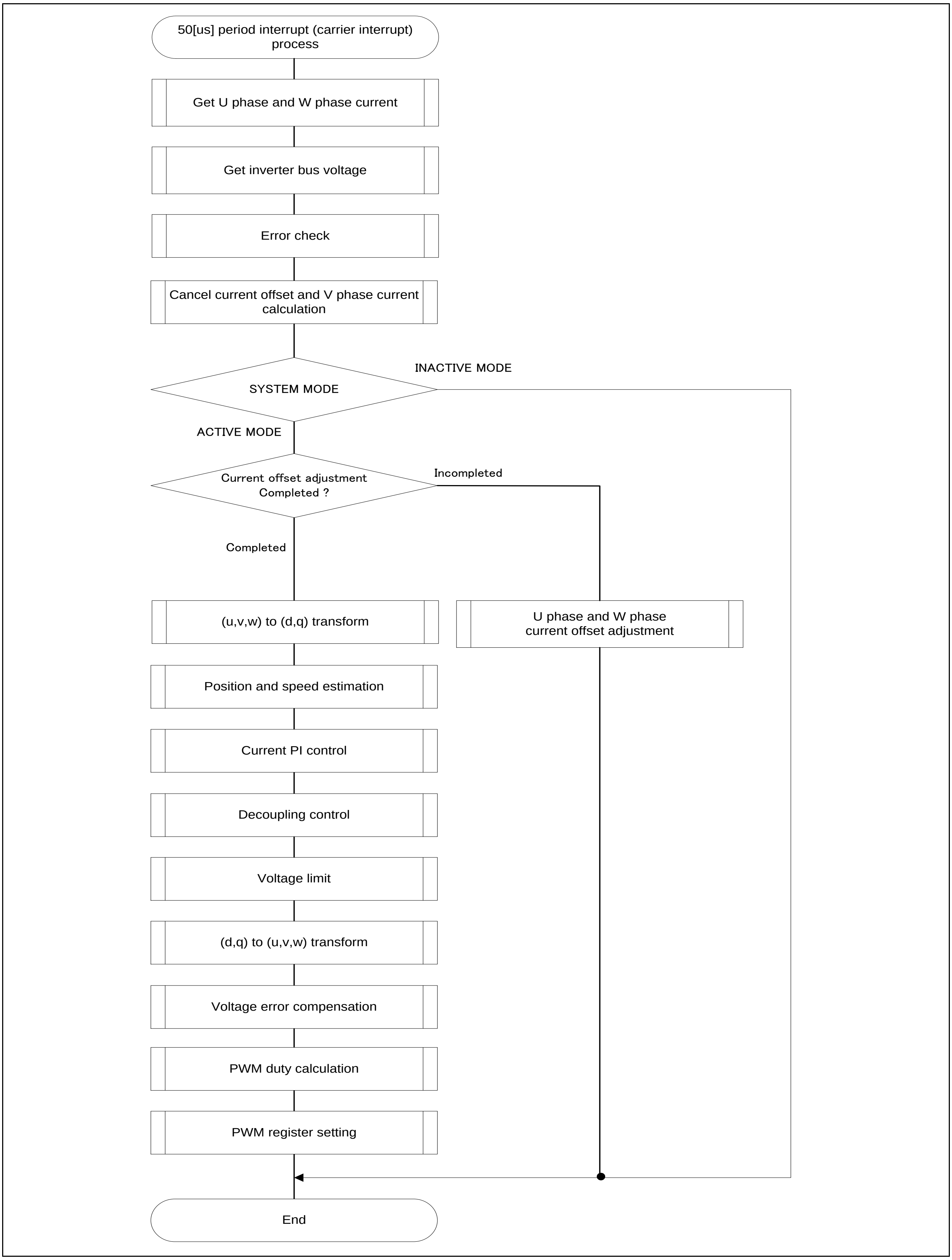


Figure 3-8 50[μs] Period Interrupt (Carrier Interrupt) Process Flowchart

3.4.3 500 [µs] Period Interrupt Process

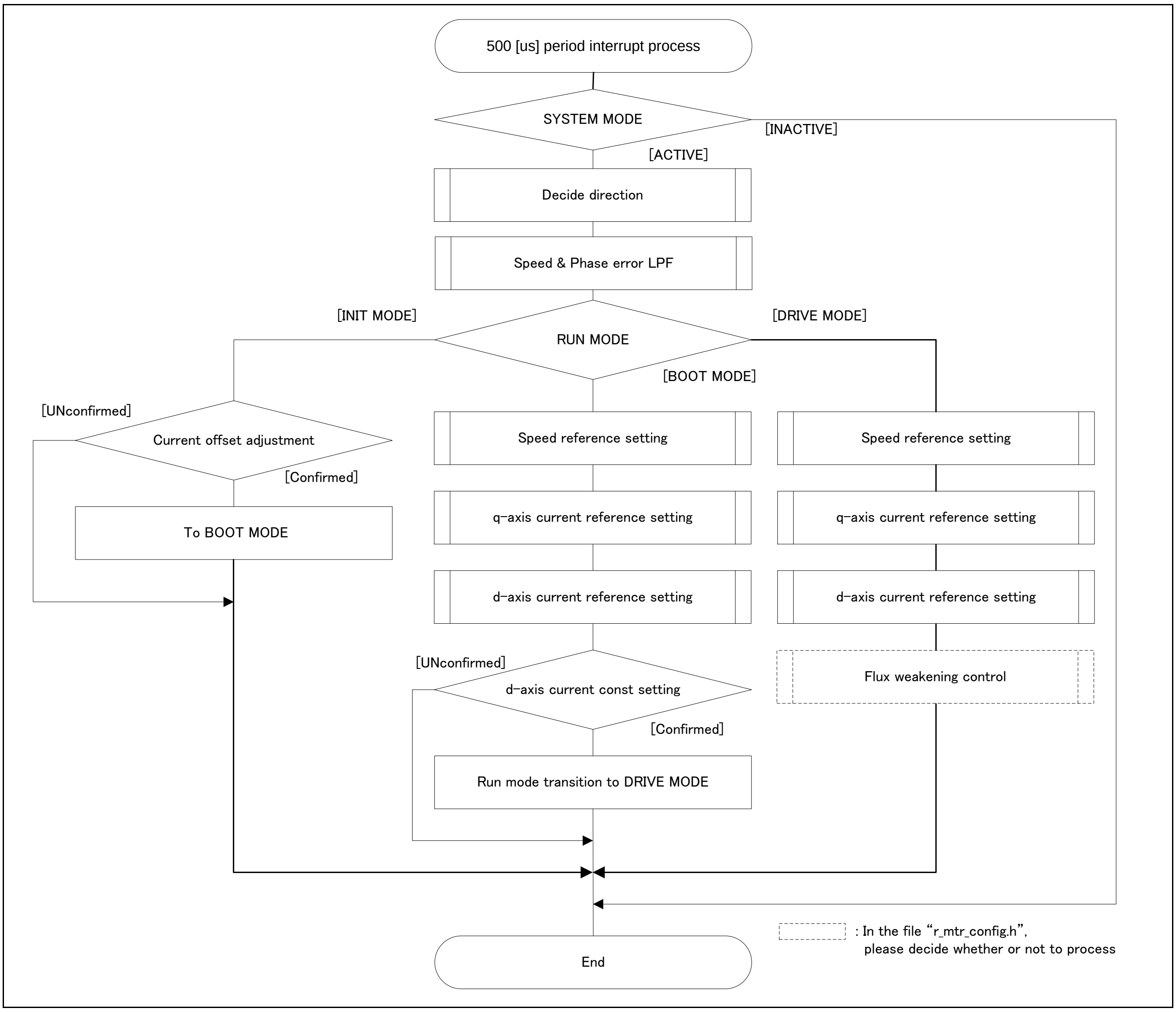


Figure 3-9 500[µs] Period Interrupt Process Flowchart

3.4.4 Over-Current Detection Interrupt Process

The over-current detection interrupt occurs when POE0# pin detects falling-edge or when output levels of the MTU complementary PWM output pins are compared and simultaneous active-level output continues for one cycle or more. Therefore, when this interrupt process is executed, PWM output pins are already in high-impedance state and the output to the motor is stopped.

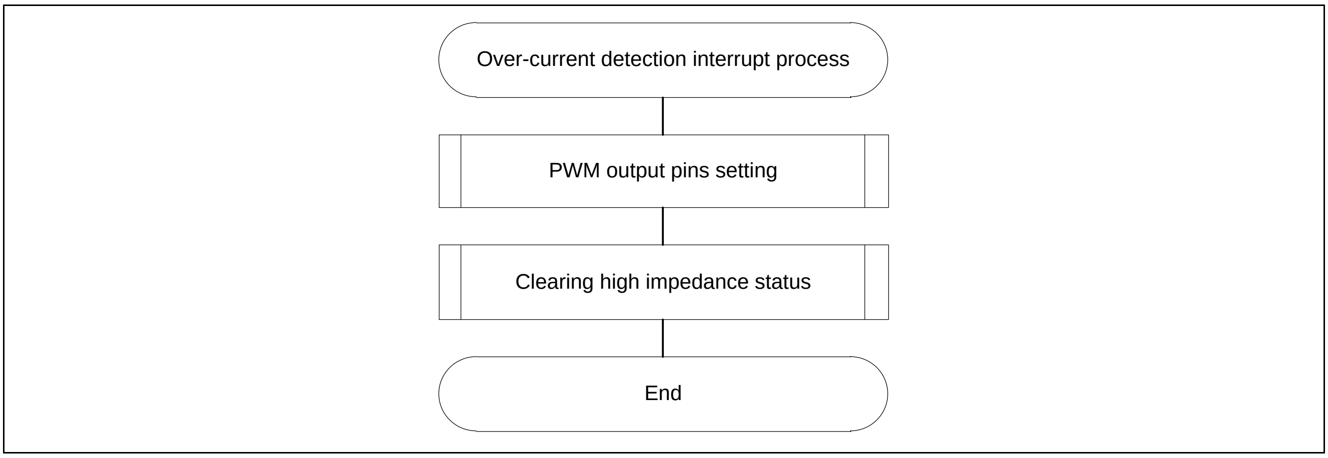


Figure 3-10 Over-Current Detection Interrupt Process Flowchart

4. Motor Control Development Support Tool 'Renesas Motor Workbench'

4.1 Overview

'Renesas Motor Workbench' is support tool for development of motor control system. 'Renesas Motor Workbench' can be used with target software of this application note to analyze the control performance. The user interfaces of 'Renesas Motor Workbench' provide functions like rotating/stop command, setting rotation speed reference, etc... Please refer to 'Renesas Motor Workbench User's Manual' for usage and more details. 'Renesas Motor Workbench' can be downloaded from Renesas Electronics Corporation website.

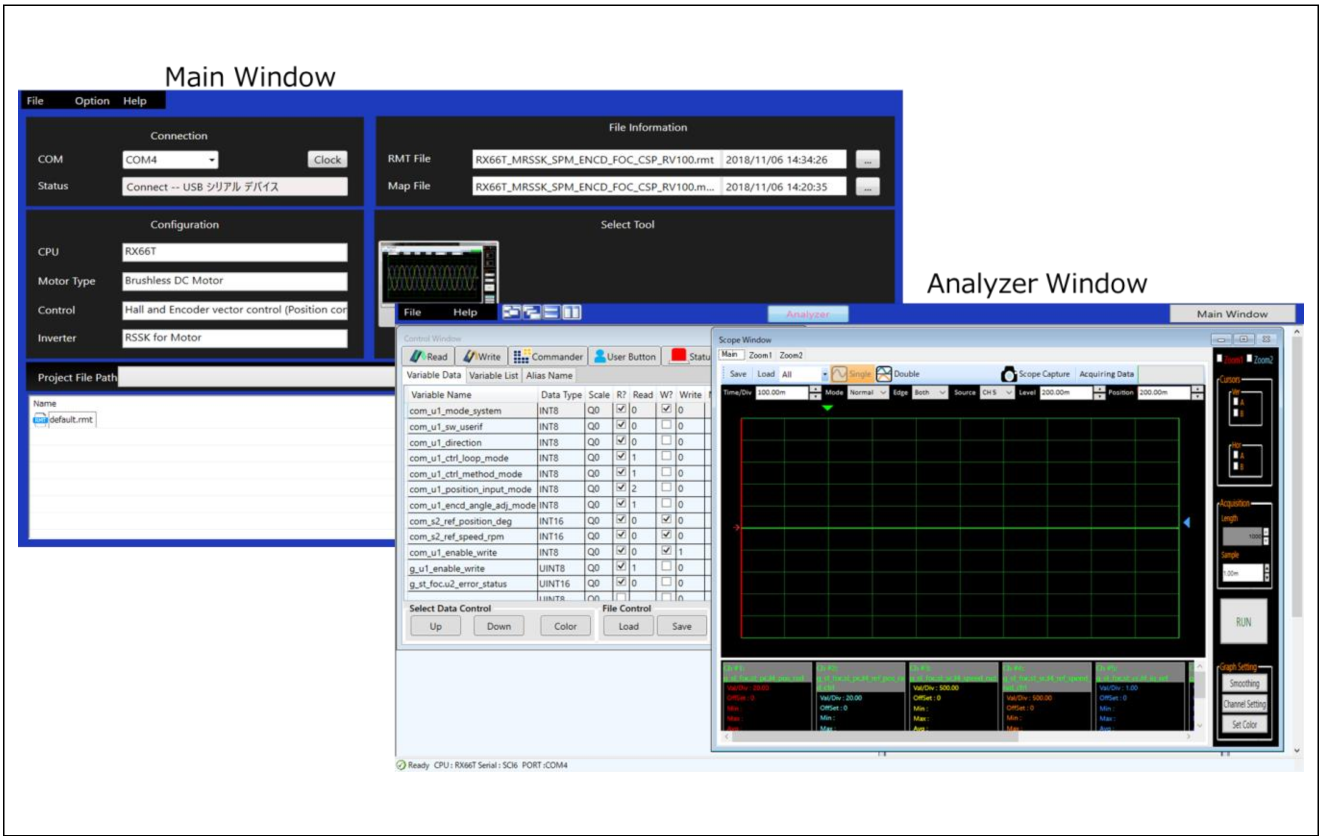
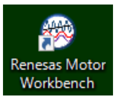


Figure 4-1 Renesas Motor Workbench – Appearance

<u>Set up for 'Renesas Motor Workbench'</u>

(1) Start 'Renesas Motor Workbench' by clicking this icon. 

(2) Click on [File] and select [Open RMT File(O)] from drop down Menu.
Select the RMT file from following location of e2studio/CS+ project folder.

'[Project Folder]/ application/user_interface/ics/'

(3) Use the 'Connection' [COM] select menu to choose the COM port.

(4) Click on the 'Analyzer' icon of Select Tool panel to open Analyzer function window.

(5) Please refer to '4.3 Operation Example for Analyzer' for motor driving operation.

4.2 List of Variables for Analyzer function

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

Table 4-1 List of Variables for Analyzer [1/2]

Variable name	Type	Content
com_u1_sw_userif (*)	uint8_t	User interface switch 0: ICS user interface use (default) 1: Board user interface use
com_u1_mode_system (*)	uint8_t	State management 0: Stop mode 1: Run mode 3: Reset
com_u1_direction	uint8_t	Rotation direction 0 : CW 1 : CCW
com_s2_ref_speed_rpm	int16_t	Speed reference (Mechanical) [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	Flux [Wb]
com_f4_mtr_j	float	Inertia [kgm^2]
com_u2_offset_calc_time	uint16_t	Current offset value calculation time [ms]
com_f4_limit_speed_change	float	Speed limit change rate (Electrical) [rpm]
com_u2_max_speed_rpm	uint16_t	Maximum speed value (Mechanical) [rpm]
com_u2_id_up_speed_rpm	uint16_t	Speed when start increasing d-axis current reference (Mechanical) [rpm]
com_f4_id_up_time	float	Ramping up time of d-axis current reference [ms]
com_f4_ref_id	float	d-axis current reference in open loop mode [A]
com_u2_id_down_speed_rpm	uint16_t	Speed when start decreasing d-axis current reference (Mechanical) [rpm]
com_f4_id_down_time	float	Decreasing time of d-axis current reference [ms]
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_current_omega	float	Natural frequency of current control system [Hz]
com_f4_current_zeta	float	Damping ratio of current 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_id_kp	float	d-axis current PI control proportional gain
com_f4_id_ki	float	d-axis current PI control Integral gain
com_f4_iq_kp	float	q-axis current PI control proportional gain
com_f4_iq_ki	float	q-axis current PI control Integral gain
com_f4_speed_kp	float	Speed PI control proportional gain
com_f4_speed_ki	float	Speed PI control Integral gain
com_u2_overspeed_limit_rpm	uint16_t	Over-speed limit value (Mechanical) [rpm]

Table 4-2 List of Variables for Analyzer [2/2]

com_f4_nominal_current_rms	float	Nominal current [A(rpm)]
com_f4_switch_phase_err_deg	float	Phase error enabled switching to sensorless control (Electrical) [deg]
com_f4_opl2less_sw_time	float	Process time of sensorless switching control [s]
com_f4_ed_hpf_omega	float	d-axis BEMF HPF cut-off frequency [Hz]
com_f4_ol_damping_zeta	float	Damping ratio of open-loop damping control
com_f4_ol_damping_fb_limit_rate	float	Feedback limit of open-loop damping control
com_f4_phase_err_lpf_cut_freq	float	Phase error LPF cut-off frequency [Hz]
com_u1_enable_write	uint8_t	Enable to rewriting variables (when the same values as of g_u1_enable_write is written)

Next, the primary variables that are frequently observed during the motor driving evaluation are listed in Table 4-3. Please refer when using Analyzer function. Regarding variables not listed in Table 4-3, refer to source codes.

Table 4-3 List of Primary Variables for Sensorless Vector Control

Name of primary variables	Type	Content
g_st_foc.st_cc.f4_id_ref	float	d-axis current reference [A]
g_st_foc.st_cc.f4_id_ad	float	d-axis current [A]
g_st_foc.st_cc.f4_iq_ref	float	q-axis current reference [A]
g_st_foc.st_cc.f4_iq_ad	float	q-axis current [A]
g_st_foc.f4_iu_ad	float	U phase current A/D conversion value [A]
g_st_foc.f4_iv_ad	float	V phase current A/D conversion value [A]
g_st_foc.f4_iw_ad	float	W phase current A/D conversion value [A]
g_st_foc.st_cc.f4_vd_ref	float	d-axis output voltage reference [V]
g_st_foc.st_cc.f4_vq_ref	float	q-axis output voltage reference [V]
g_st_foc.f4_refu	float	U phase voltage reference [V]
g_st_foc.f4_refv	float	V phase voltage reference [V]
g_st_foc.f4_refw	float	W phase voltage reference [V]
g_st_foc.f4_modu	float	U phase modulation factor
g_st_foc.f4_modv	float	V phase modulation factor
g_st_foc.f4_modw	float	W phase modulation factor
g_st_foc.f4_ed	float	Estimated d-axis BEMF [V]
g_st_foc.f4_eq	float	Estimated q-axis BEMF [V]
g_st_foc.st_rotor_angle.f4_rotor_angle_rad	float	Estimated position (Electrical) [rad]
g_st_foc.st_sc.f4_ref_speed_rad_ctrl	float	Speed reference (Electrical) [rad/s]
g_st_foc.st_sc.f4_speed_rad	float	Estimated speed (Electrical) [rad/s]
g_st_foc.f4_phase_err_rad	float	Phase error (Electrical) [rad]
g_st_foc.u2_error_status	uint16_t	Error status

4.3 Operation Example for Analyzer

The section shows an example below for motor driving operation using Analyzer. Operation is using ‘Control Window’ of Analyzer. Regarding specification of ‘Control Window’, refer to ‘Renesas Motor Workbench User’s Manual’.

- Driving the motor

(1) Confirm the check-boxes of column [W?] for ‘com_u1_mode_system’, ‘com_s2_ref_speed_rpm’, ‘com_u1_enable_write’ marks.

(2) Input a reference speed value in the [Write] box of ‘com_s2_ref_speed_rpm’.

(3) Click the ‘Write’ button.

(4) Click the ‘Read’ button. Confirm the [Read] box of ‘com_s2_ref_speed_rpm’, ‘g_u1_enable_write’.

(5) Set a same value of ‘g_u1_enable_write’ in the [Write] box of ‘com_u1_enable_write’.

(6) Write ‘1’ in the [Write] box of ‘com_u1_mode_system’.

(7) Click the ‘Write’ button.

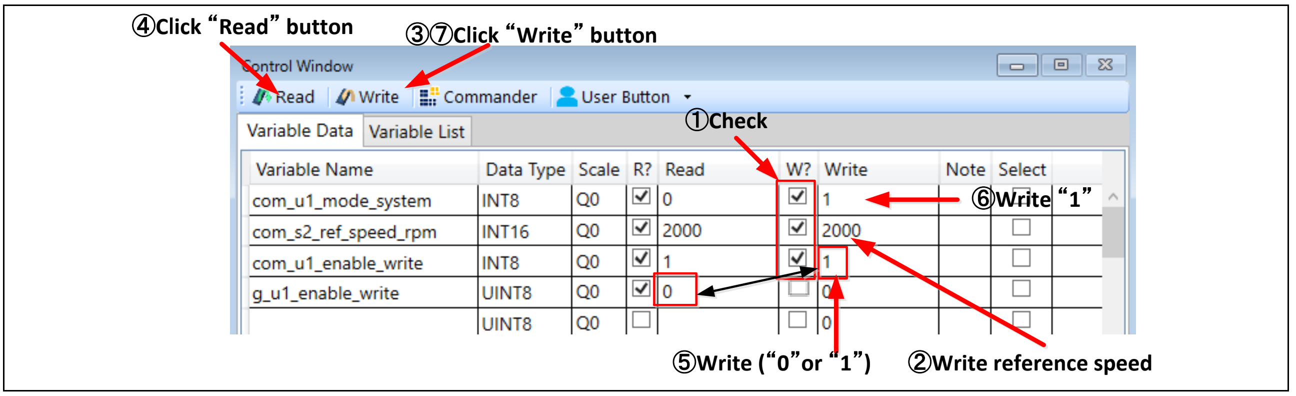


Figure 4-2 Procedure - Driving the Motor

- Stop the motor

(1) Write ‘0’ in the [Write] box of ‘com_u1_mode_system’

(2) Click the ‘Write’ button.

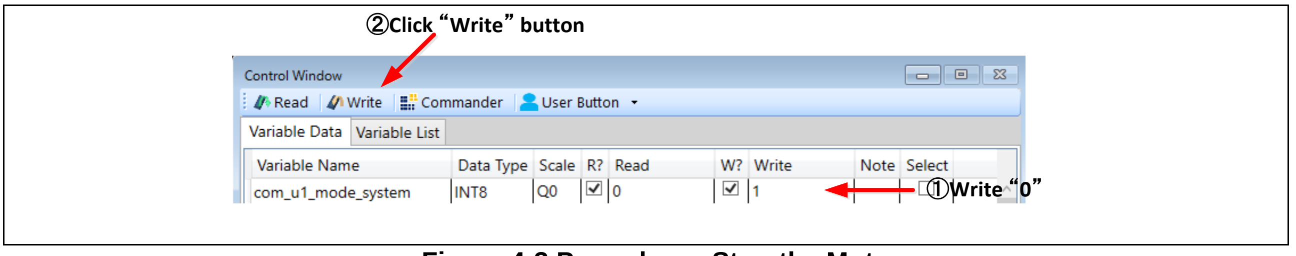


Figure 4-3 Procedure - Stop the Motor

- Error cancel operation

(1) Write ‘3’ in the [Write] box of ‘com_u1_mode_system’

(2) Click the ‘Write’ button.

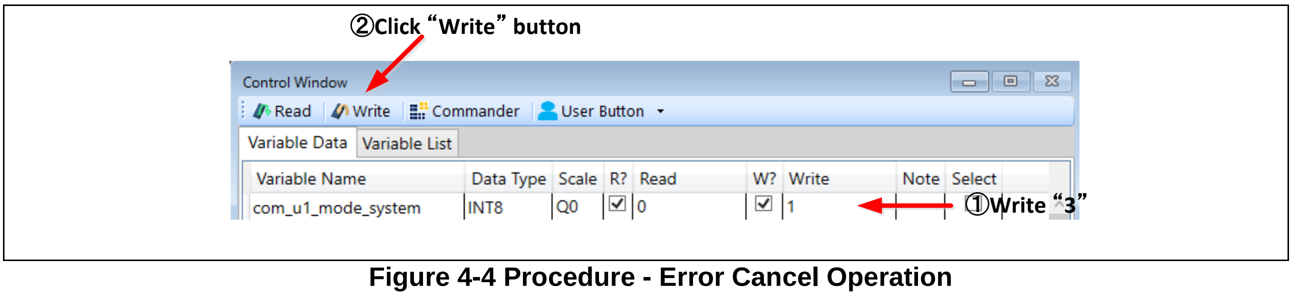


Figure 4-4 Procedure - Error Cancel Operation

4.4 Operation Example for User Button

The section shows an example below for motor driving operation using User Button.

- Driving or Stop the motor

By setting as shown in Figure 4-5, driving and stopping change each time the button is pressed.

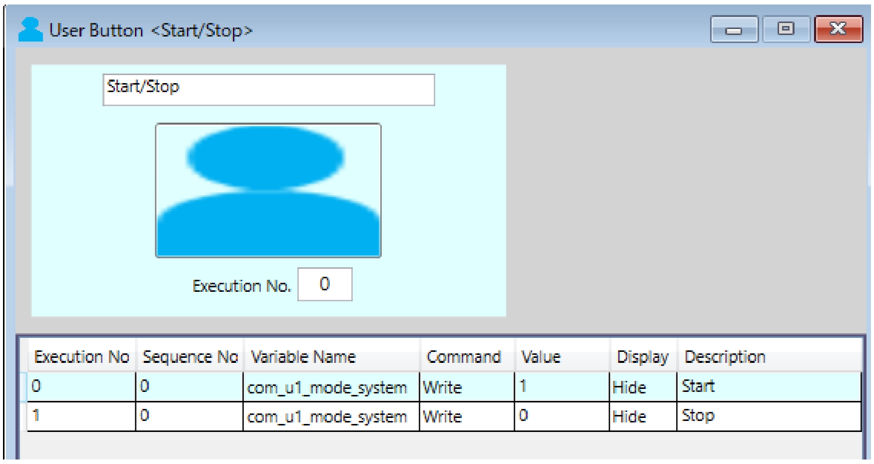


Figure 4-5 Driving or Stop the Motor

- Change speed

By setting as shown in Figure 4-6, enter the command speed and press the button to change the speed.

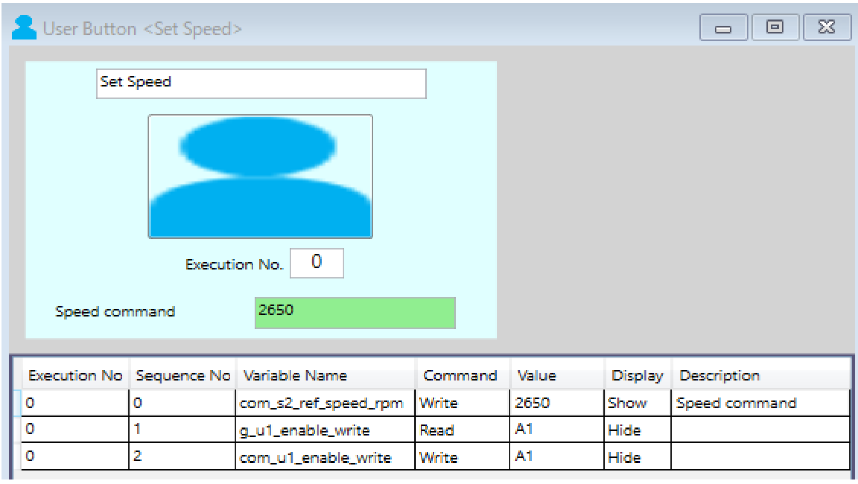


Figure 4-6 Change speed

Website and Support

Renesas Electronics Website
http://www.renesas.com/

Inquiries
http://www.renesas.com/contact/

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Mar. 29, 2019	-	First edition issued

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.

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

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/.