Vector Control for Permanent Magnet Synchronous Motor with Magnet sensor (Implementation)

RX24T

Abstract

This application note aims to explain the sample programs for a permanent magnet synchronous motor with magnet sensor, by using functions of RX24T. The explanation includes, how to use the library of 'Renesas Motor Workbench' tool, that is a 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.

This application note describes motor control using magnetic sensors whose angle information output format is digital output (AB pulse) and analog output (sine wave, cos wave). The specifications for each output format will be explained in this application, but when using motor control with digital output magnetic sensor, use the vector control software that uses the encoder described in the reference material.

Operation Checking Device

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

- RX24T (R5F524TAADFP)

Target Software

The target programs of this application note are as follows.

- RX24T_MRSSK_SPM_MAG_FOC_CSP_RV100 (IDE: CS+)
- RX24T_MRSSK_SPM_MAG_FOC_E2S_RV100 (IDE: e^{2}studio)
  RX24T vector control with magnet sensor software for '24V Motor Control Evaluation System for RX23T and RX24T CPU CARD'
- RX24T_MRSSK_SPM_ENCD_FOC_CSP_RV101 (IDE: CS+)
- RX24T_MRSSK_SPM_ENCD_FOC_E2S_RV101 (IDE: e^{2}studio)
  RX24T vector control with encoder software for '24V Motor Control Evaluation System for RX23T and RX24T CPU CARD'

Reference

- RX24T Group User's Manual: Hardware (R01UH0576EJ0200)
- Application note: 'Vector control for permanent magnet synchronous motor with encoder (Algorithm)' (R01AN3789EJ0102)
- Application note: 'Vector control for permanent magnet synchronous motor with encoder (Imprementation)' (R01AN3791EJ0101)
- Renesas Motor Workbench V.2.00 User's Manual (R21UZ0004EJ0202)
- Renesas Solution Starter Kit 24V Motor Control Evaluation System for RX23T User's Manual (R20UT3697EJ0100)
- RX24T CPU CARD User's Manual (R20UT3696EJ0110)

Contents

1. Overview

This application note aims to explain the sample programs for a permanent magnet synchronous motor with magnet sensor, by using functions of RX24T. The explanation includes, how to use the library of 'Renesas Motor Workbench' tool, that is a support tool for motor control development.

1.1 Development Environment

Table 1-1 and Table 1-2 shows the development environment of the sample programs explained in this application note.

Table 1-1 Hardware Development Environment

Microcontroller	Evaluation board	Motor
RX24T (R5F524TAADFP)	24V inverter circuit & RX24T CPU Card*1	iSS57-20*2, *3

Table 1-2 Software Development Environment

CS+ version	e^2studio version	Toolchain version
V8.03.00	V7.7.0	CC-RX V3.02.00

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

Notes: 1. 24V inverter board & RX24T CPU Card (RTK0EM0009C03402BJ) are products of Renesas Electronics Corporation.

2. iSS57-20 motor is a product of Leadshine Technology. The operation has been confirmed at the condition of the iSS57-20 motor assembled with the sensor board using the TMR magnetic sensors TAD2141 and TAS2143 manufactured by TDK Corporation and the magnet for magnetic field detection. Please contact to TDK Corporation for the TMR sensor board and the detection magnet.
Leadshine Technology Co., Ltd(http://www.leadshine.com/)
TDK Corporation (https://product.tdk.com/info/ja/products/sensor/angle/tmr-angle/index.html)

3. In this operating environment, the magnet for detection of 1 pole pair is used for the motor of 50 pole pairs, so in the environment where TAS2143 is used, the detection angle error becomes large, so position control cannot be performed. When performing position control, match the number of pole pairs of the motor and sensor detection magnets.

2. System Overview

Overview of this system is explained below.

2.1 Hardware Configuration

The hardware configuration is shown below.

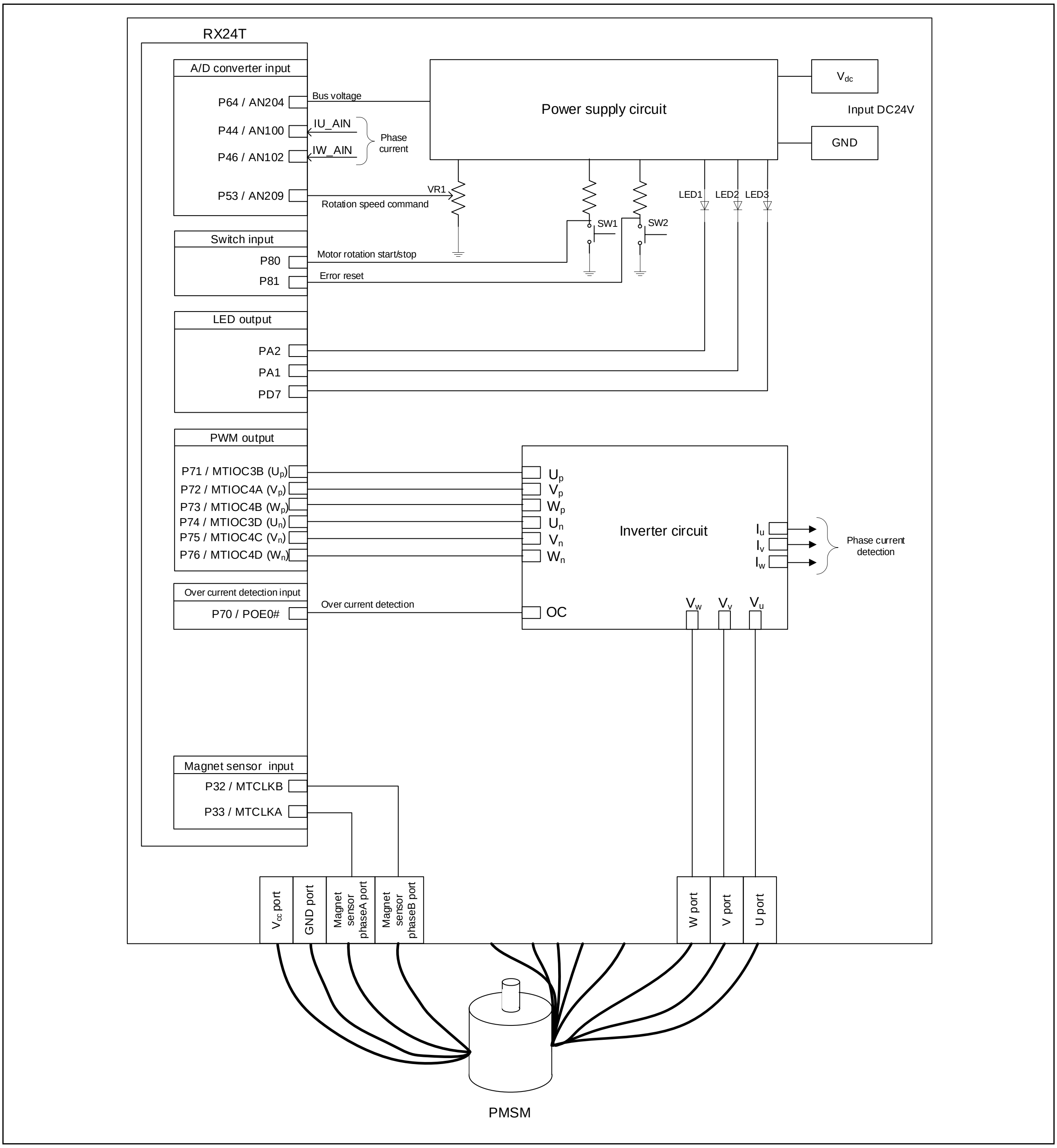


Figure 2-1 Hardware Configuration Diagram (Used at TAD2141)

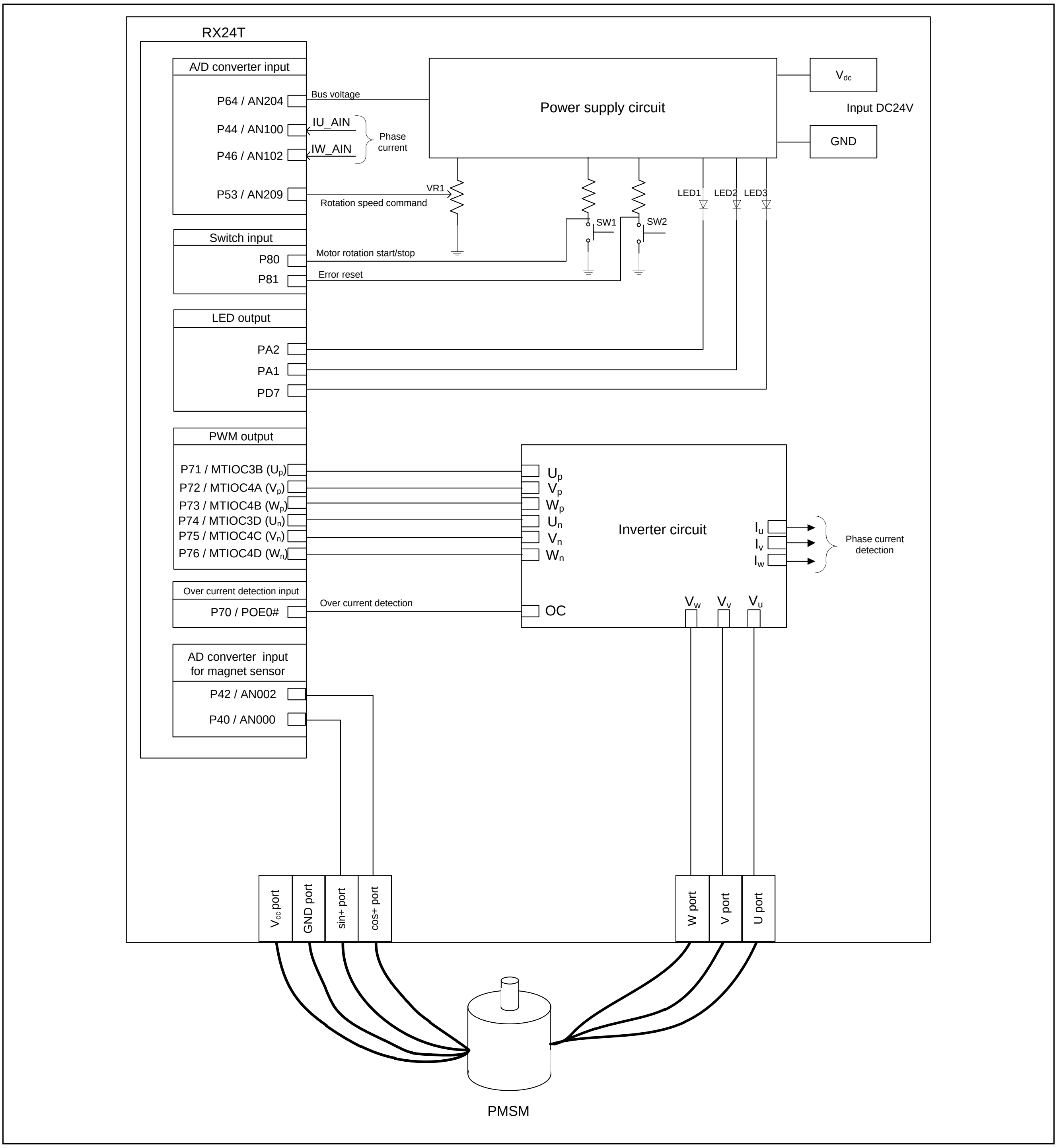


Figure 2-2 Hardware Configuration Diagram (Used at TAS2143)

2.2 Hardware Specifications

2.2.1 User Interfaces

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

Table 2-1 User Interfaces

Item	Interface component	Function
Rotation position / speed	Variable resistor (VR1)	Reference value of rotation position / 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
LED3	Yellow green LED	• Complete of positioning: ON • Uncomplete of positioning: OFF
RESET	Push switch (RESET1)	System reset

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

Table 2-2 Port Interfaces

R5F524TAADFP Port name	Function
P64 / AN204	Inverter bus voltage measurement
P53 / AN209	For position / speed command value input (analog value)
P80	START/STOP toggle switch
P81	ERROR RESET toggle switch
PA2	LED1 ON/OFF control
PA1	LED2 ON/OFF control
PD7	LED3 ON/OFF control
P44 / AN100	U phase current measurement
P46 / AN102	W phase current measurement
P71 / MTIOC3B	PWM output (U_p)
P72 / MTIOC4A	PWM output (V_p)
P73 / MTIOC4B	PWM output (W_p)
P74 / MTIOC3D	PWM output (U_n)
P75 / MTIOC4C	PWM output (V_n)
P76 / MTIOC4D	PWM output (W_n)
P40 / AN000	Magnet sensor sin+ signal input (TAS2143)
P42 / AN002	Magnet sensor cos+ signal input (TAS2143)
P32 / MTCLKB	Magnet sensor Phase-B signal input (TAD2141)
P33 / MTCLKA	Magnet sensor Phase-A signal input (TAD2141)
P70 / POE0#	PWM emergency stop input for 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	CMT	MTU3	POE3
• Rotation speed command value input • Current of each phase U and W measurement • Inverter bus voltage measurement • Magnet sensor output	1 [ms] interval timer	• Complementary PWM output • Magnet sensor phase counter • Magnet sensor count capture	Set PWM output ports to high impedance state to stop the PWM output.

(1) 12-bit A/D converter (S12ADE)

U phase current (I_u), W phase current (I_w), inverter bus voltage (V_{dc}) 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 (I_u) and W phase current (I_w) measurement.

(2) Compare match timer (CMT)

Channel 0 of the compare match timer is used for 1 [ms] interval timer.

(3) Multi-function timer pulse unit 3 (MTU3)

The operation mode varies depending on channels. On the channels 3 and 4, output (active level: high) with dead time is performed by using the complementary PWM mode.

Channel 1 of MTU3 operate in phase counting mode, the counter is incremented or decremented according to the phase difference between Phase-A and Phase-B signals from the encoder.

Channel 0 of MTU3 is used as free-run timer for speed measurement of digital magnet sensor. For details, refer to the application note of “Encoder vector control of permanent magnet synchronous motor (implementation)”.

(4) Port output enable 3 (POE3)

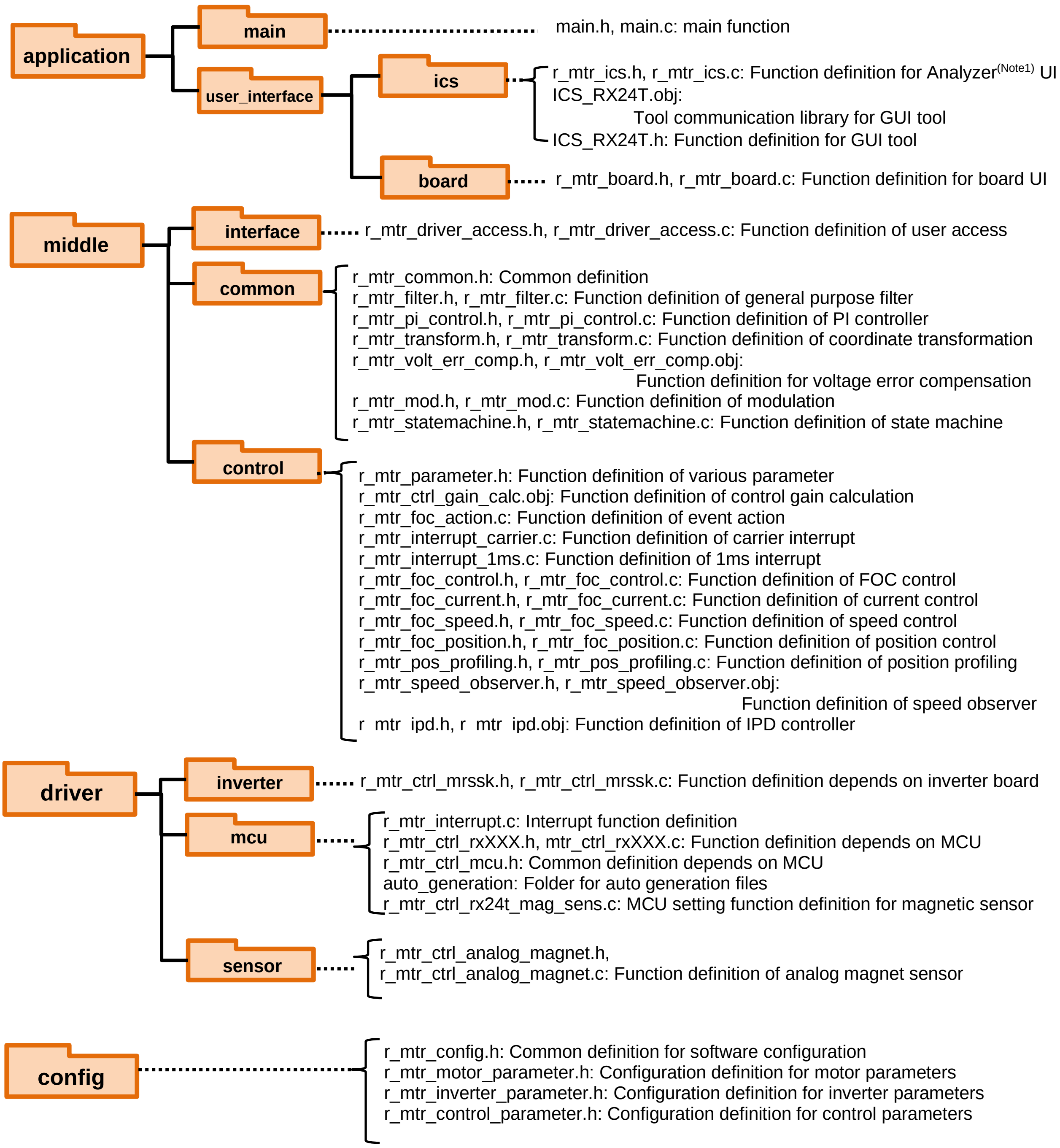
PWM output ports are set to high impedance state when an overcurrent is detected (when a falling edge of the POE0# port is detected) and when an output short circuit is detected.

2.3 Software Configuration

2.3.1 Software File Configuration

Folder and file configuration of the sample programs is 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 2-3 Folder and File Configuration

2.3.2 Module Configuration

Module configuration of the sample programs is described below.

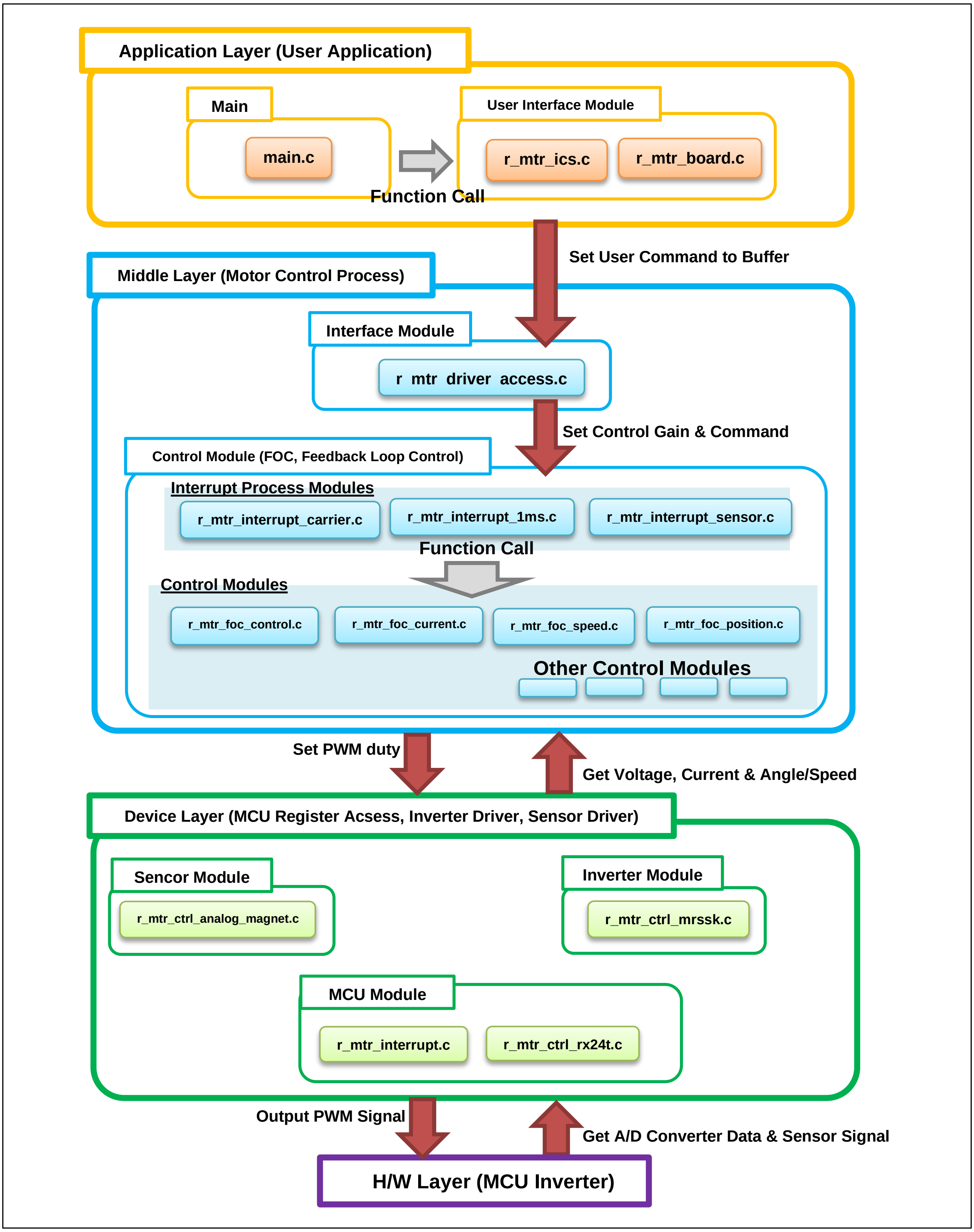


Figure 2-4 Module Configuration

2.4 Software Specifications

Table 2-4 shows the basic software specification of this system. For details of the vector control, refer to the application note 'Vector control of permanent magnet synchronous motor: algorithm'.

Table 2-4 Basic Specifications of Vector Control PMSM with Encoder 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>Motor position control start/stop</td><td colspan="2">Determined depending on the level of SW1 ('Low': control start, 'High': stop) or input from Renesas Motor Workbench</td></tr>
<tr><td>Position detection of rotor magnetic pole</td><td colspan="2">Magnet sensor (Digital: A-B Phase), Magnet sensor (Analog: sin+, cos+)</td></tr>
<tr><td>Input voltage</td><td colspan="2">DC 24 [V]</td></tr>
<tr><td>Main clock frequency</td><td colspan="2">80 [MHz]</td></tr>
<tr><td>Carrier frequency (PWM)</td><td colspan="2">20 [kHz] (carrier cycle: 50 [μs])</td></tr>
<tr><td>Dead time</td><td colspan="2">2 [μs]</td></tr>
<tr><td>Control cycle
(Current loop)</td><td colspan="2">100 [μs] (twice the carrier cycle)</td></tr>
<tr><td>Control cycle
(Speed and Position loop)</td><td colspan="2">1 [ms]</td></tr>
<tr><td rowspan="2">Management of position command value</td><td>Board UI</td><td>Position command generation: Direct input of VR1
(input range)
-180° to 180°</td></tr>
<tr><td>ICS UI</td><td>Position command generation: Position profile of trapezoidal curve for speed command value
(input range)
- 32768° to 32767°
(Max speed)
CW / CCW: 800 [rpm]</td></tr>
<tr><td>Management of speed command value</td><td colspan="2">CW: 0 [rpm] to 500 [rpm]
CCW: 0 [rpm] to 500 [rpm]</td></tr>
<tr><td>Accuracy of position</td><td colspan="2">Digital: 16384 [ppr], Analog: (Sensor out put [V] / 5 [V]) × 4096 [ppr]</td></tr>
<tr><td>Dead band of position</td><td colspan="2">Digital: count ±1 [cpr], Analog: ±3 [degree]
Note: Dead zone is provided to prevent hunting in positioning.</td></tr>
<tr><td>Natural frequency of each control system</td><td colspan="2">Current control system: 200 Hz
Speed control system: 20 Hz
Position control system: 5 Hz</td></tr>
<tr><td rowspan="2">Optimization setting for compiler</td><td>Optimization level</td><td>2 (-optimize = 2) (default)</td></tr>
<tr><td>Optimization method</td><td>Size priority (default)</td></tr>
<tr><td>ROM/RAM size</td><td colspan="2">ROM: 17.4 KB
RAM: 4.6 KB</td></tr>
<tr><td>Sensor output correct function</td><td colspan="2">Enable/disable switchable: Corrects analog output signal of magnetic sensor.</td></tr>
<tr><td>Processing stop for protection</td><td colspan="2">Motor control signal outputs (six outputs) will be disabled, under any of the following conditions.
1. Current of each phase exceeds 3.82 [A] (monitored every 100 [μs])
2. Inverter bus voltage exceeds 28 [V] (monitored every 100 [μs])
3. Inverter bus voltage is less than 14 [V] (monitored every 100 [μs])
4. Rotation speed exceeds 800 [rpm] (monitored every 100 [μs])

When an external over-current signal is detected (when a falling edge of the POE0# port is detected) and 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 sample programs of this application note are explained below.

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', the program determines that the motor should be stopped.

3.1.2 A/D Converter

(1) Motor Rotation Position and Speed Command Value

The motor rotation position and speed command value 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 Command Value

Item	Conversion ratio (Command value: A/D conversion value)		Channel
Rotation position command value	CW	0° to 180°: 0800H to 0FFFH	AN209
	CCW	0° to -180°: 07FFH to 0000H	
Rotation speed command value	CW	0 [rpm] to 500 [rpm]: 0800H to 0FFFH	
	CCW	0 [rpm] to 500 [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, under-voltage detection 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	AN204

(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 *	Iu: AN100 Iw: AN102

Note: * For more details of A/D conversion characteristics, refer to 'RX24T Group User's Manual: Hardware'.

3.1.3 Position Profile Generation (Position Profile of Trapezoidal Curve for Speed Command Value)

In vector control software for PMSM with magnet sensor, the position profile generation is used to create command value (input position value). The implementation of command value in each control cycle is used as method of managing acceleration and the maximum speed value with respect to target position value.

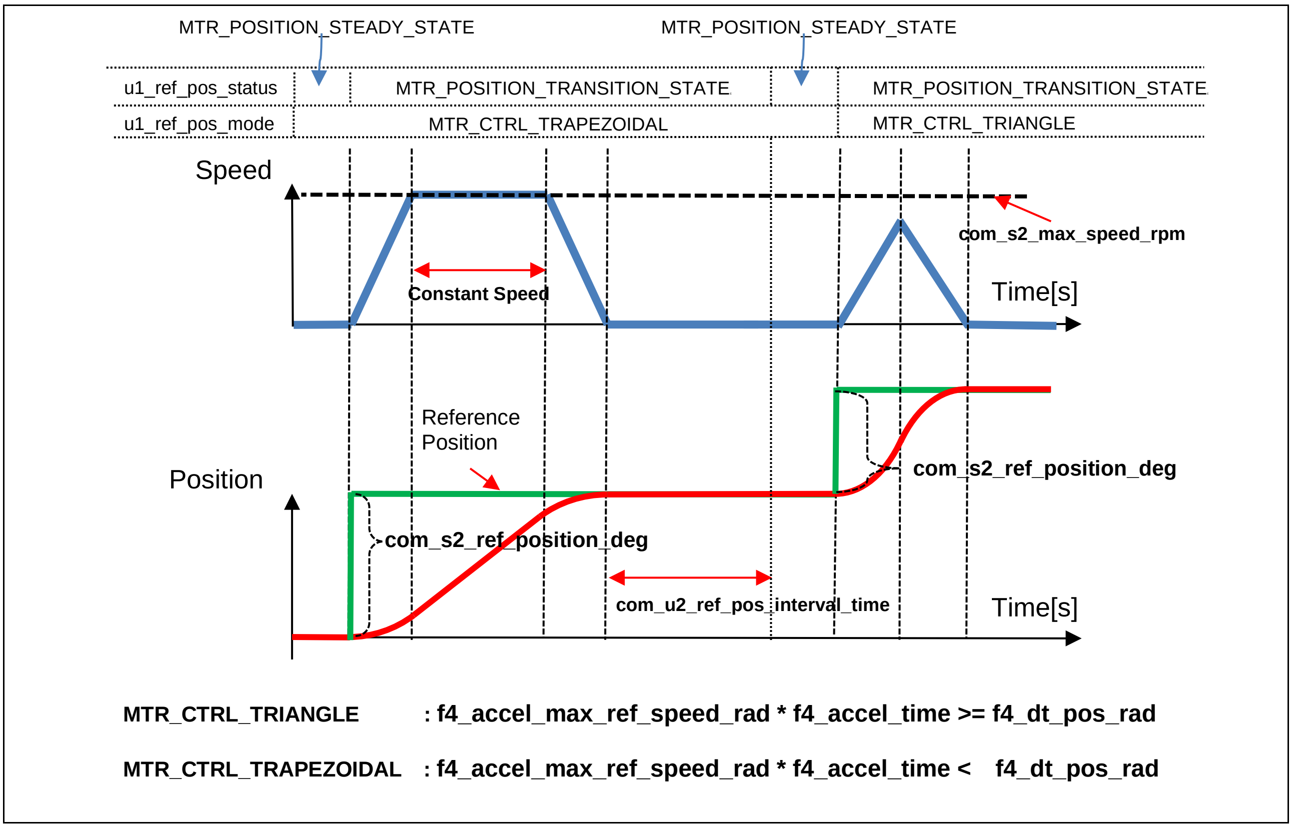


Figure 3-1 Position Profile Generation

By inputting the following variables from the Analyzer, it is possible to create command values that enable acceleration/deceleration response.

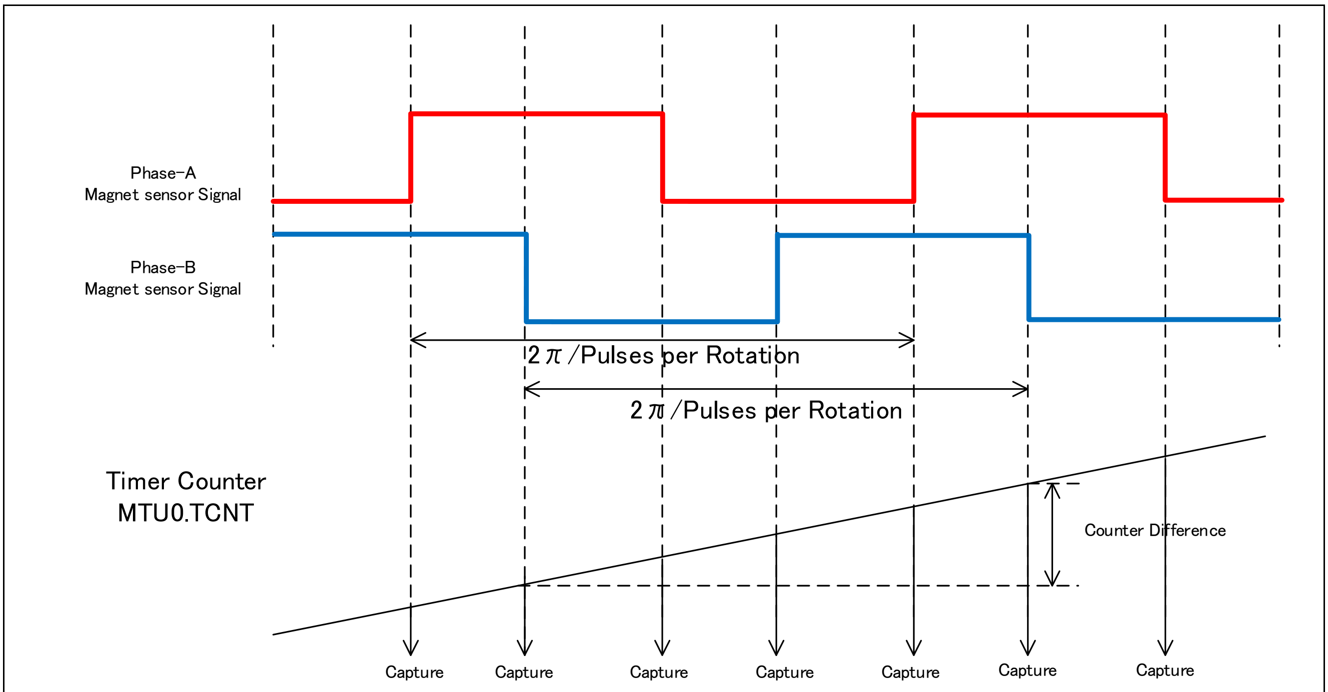
- Acceleration time (com_f4_accel_time)
- Maximum speed (com_f4_accel_max_ref_speed_rad)
- Position stabilization wait time (com_u2_ref_pos_interval_time)

When the speed calculated from the position deviation and acceleration time is higher than the maximum speed during acceleration, the trapezoidal speed command value is used.

3.1.4 Speed Measurement

In order to obtain better real-time performance and higher speed resolution at low speed, this system use magnet sensor signal edge interval to calculate speed, the speed extrapolation is used in PI control calculation. In addition, taking the difference between rise time and fall time and the accuracy of quadrature of magnet sensor signal into consideration, the speed is calculated with time elapsed and angle changed in one period of magnet sensor Phase-A or Phase-B signals.

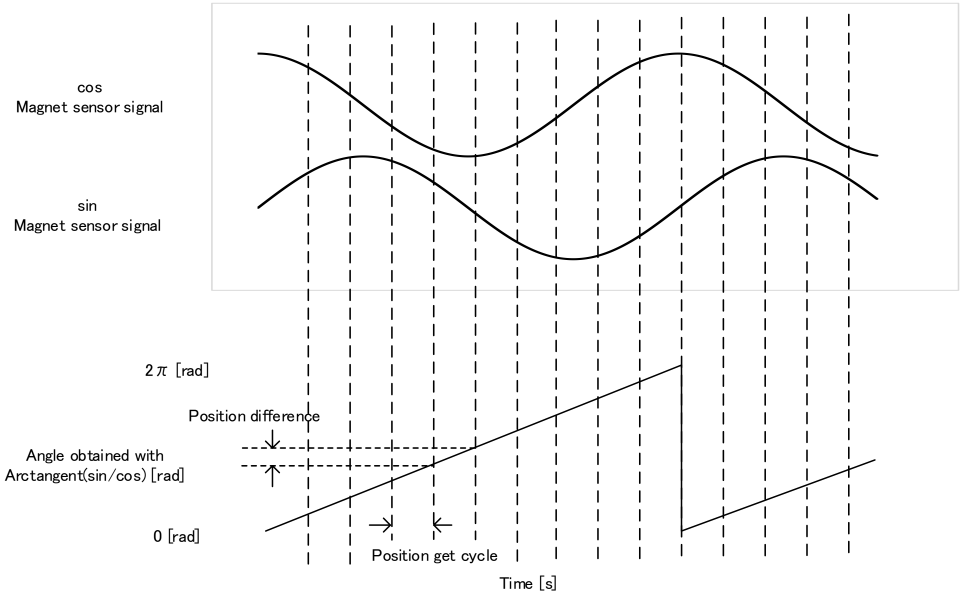
(1) Speed Calculation (Digital magnet sensor)



$$\text{Motor Rotation Speed[rad/s]} = \left(\frac{2\pi}{\text{Count Per Rotation}}\right) / \left(\frac{\text{Counter Difference}}{\text{Timer Clock Frequency}}\right)$$

Figure 3-2 Speed Calculation using digital magnet sensor

(2) Speed Calculation (Analog magnet sensor)



$$\text{Motor Rotation Speed[rad/s]} = (\text{Positon Difference})/(\text{Position get cycle})$$

Figure 3-3 Speed Calculation using Analog magnet sensor

3.1.5 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 and smaller than the carrier wave respectively.

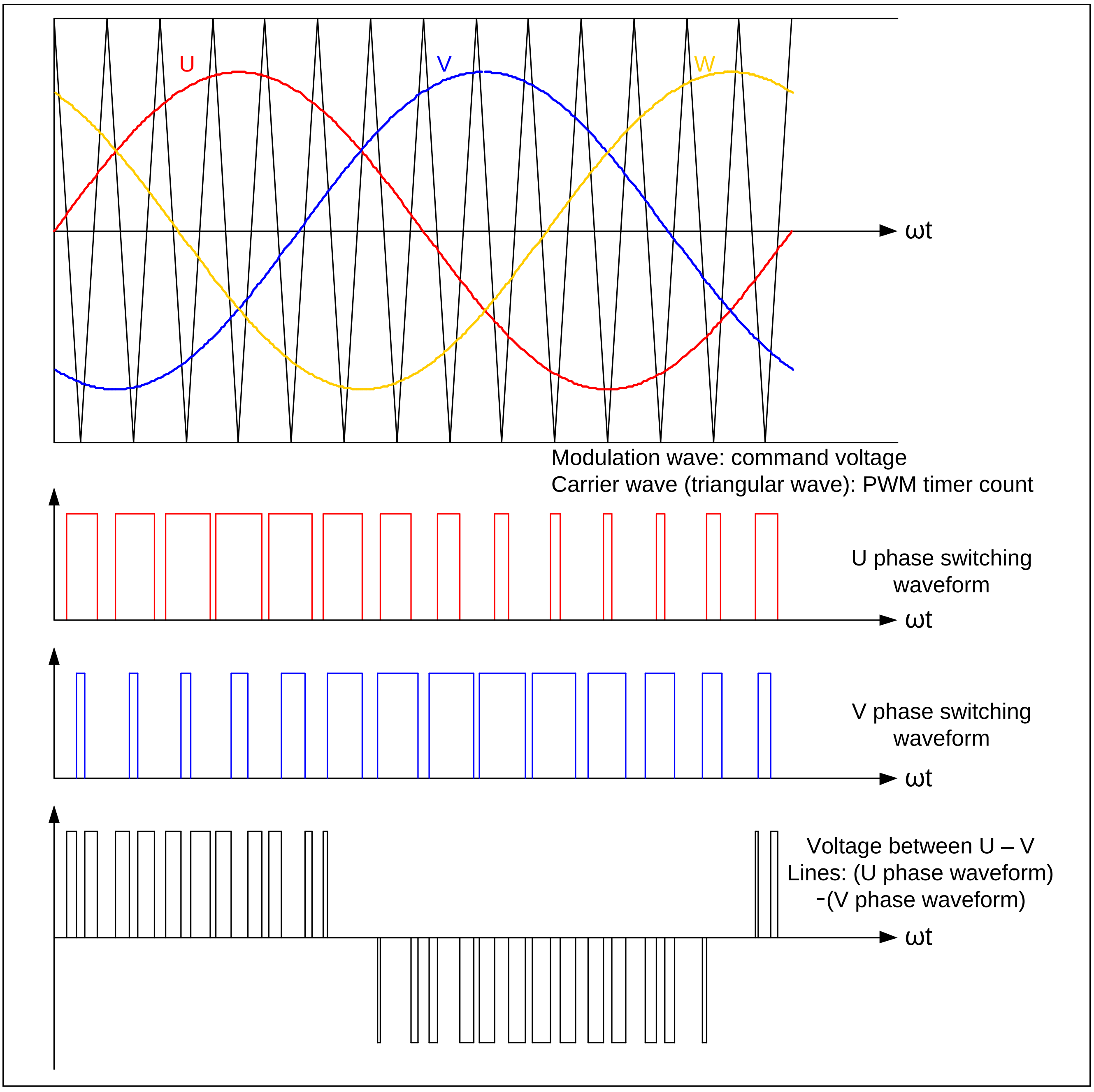


Figure 3-4 Conceptual Diagram of the Triangular Wave Comparison Method

Here, as shown in Figure 3-5, ratio of the output voltage pulse to the carrier wave is called duty.

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

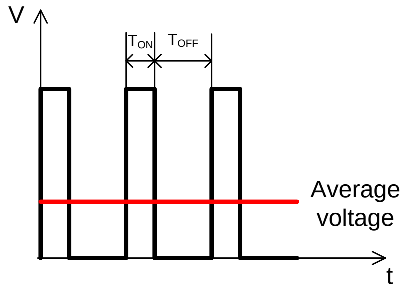


Figure 3-5 Definition of Duty

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.

3.1.6 State Transition

Figure 3-6 is a state transition diagram of the 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.

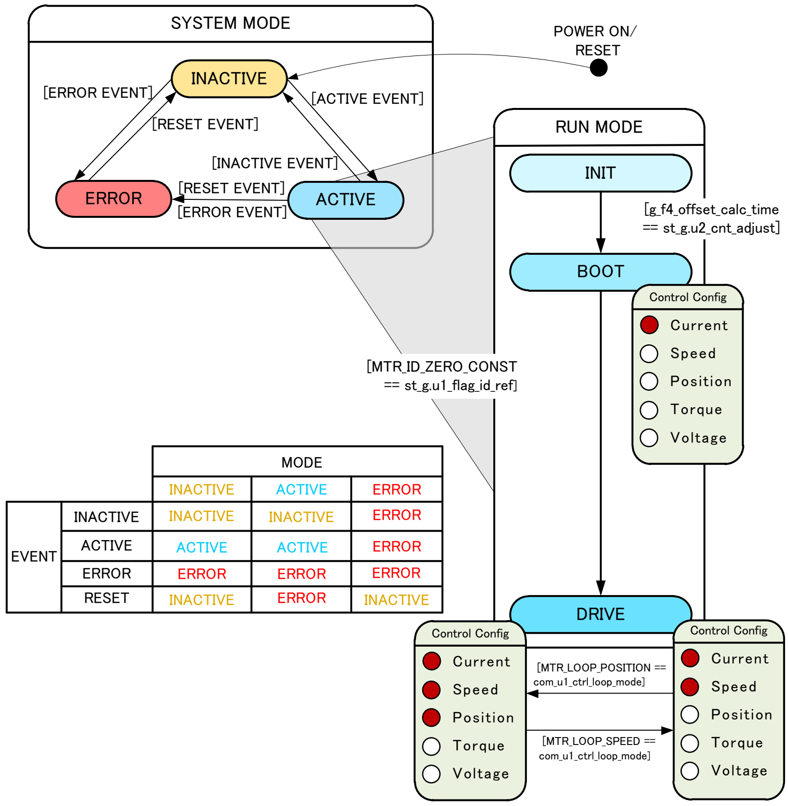


Figure 3-6 State Transition Diagram of Vector Control PMSM with magnet sensor Software

(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 3-6 when 'SYSTEM MODE' is 'ACTIVE'.

(3) EVENT

When 'EVENT' occurs in each 'SYSTEM MODE', 'SYSTEM MODE' changes as shown the table of Figure 3-6, 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

3.1.7 Startup Method

Table 3-7, Table 3-8 shows the software implementation of d-axis and magnet sensor alignment method. The d-axis alignment method used as startup control of position control method, in initialization mode (MTR_MODE_INIT) and Boot mode (MTR_MODE_BOOT). In drive mode (MTR_MODE_DRIVE) vector control is implemented for PMSM with Magnet sensor of digital output. Each reference value setting of d-axis current, q-axis current and speed is managed by respective status.

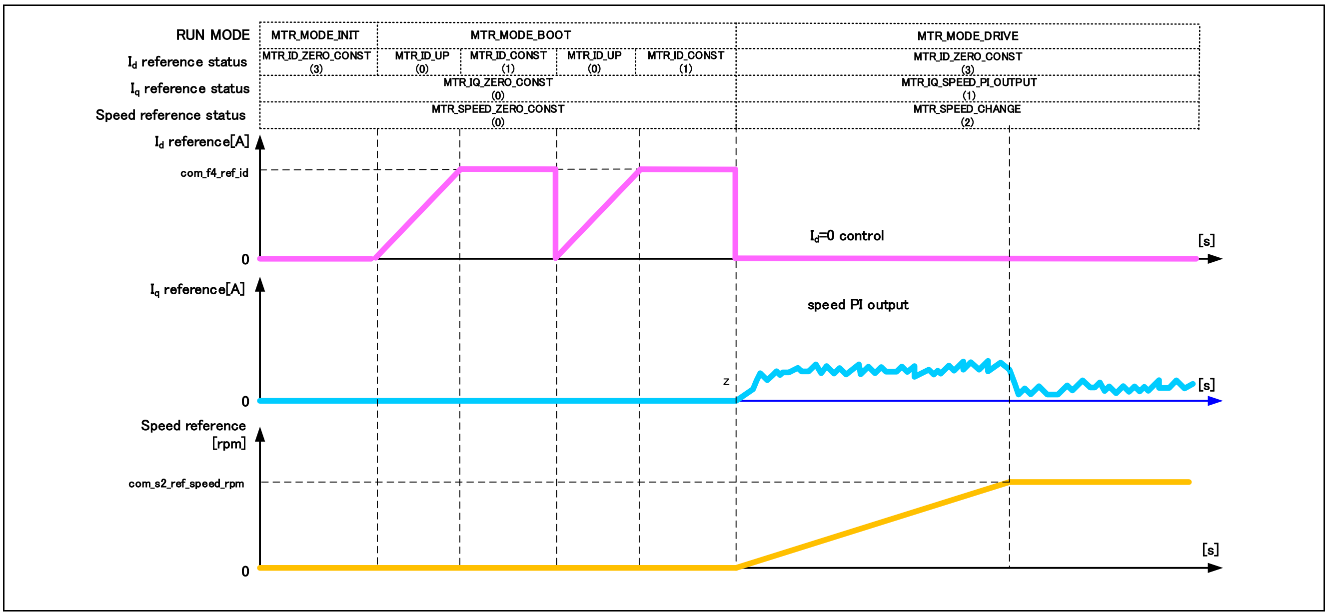


Figure 3-7 Startup Speed Control of Vector Control PMSM with digital Magnet sensor Software

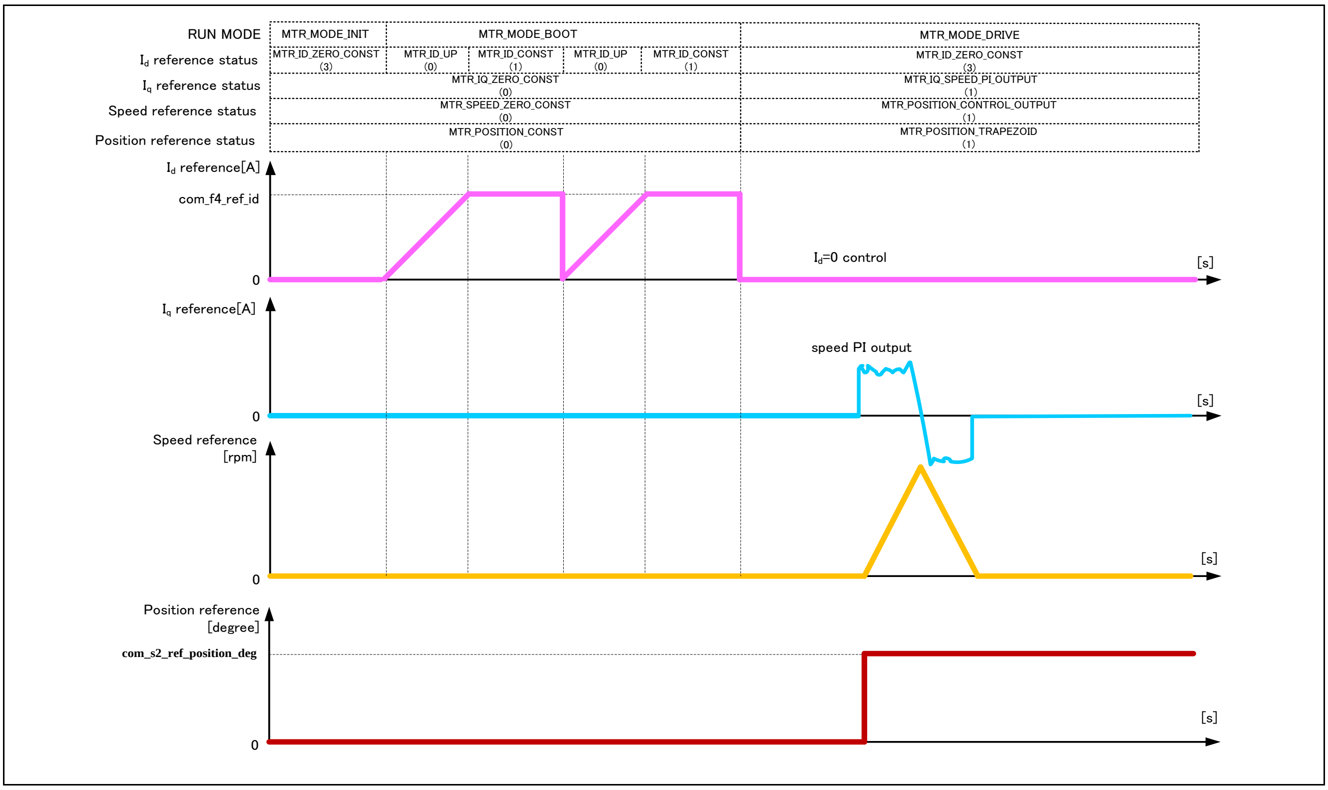


Figure 3-8 Startup Position Control of Vector Control PMSM with digital Magnet sensor Software

Table 3-9 shows the software implementation of d-axis and magnet sensor alignment method. The d-axis alignment method used as startup control of position control method, in initialization mode (MTR_MODE_INIT) and Boot mode (MTR_MODE_BOOT). In drive mode (MTR_MODE_DRIVE) vector control is implemented for PMSM with Magnet sensor of analog output. Each reference value setting of d-axis current, q-axis current and speed is managed by respective status.

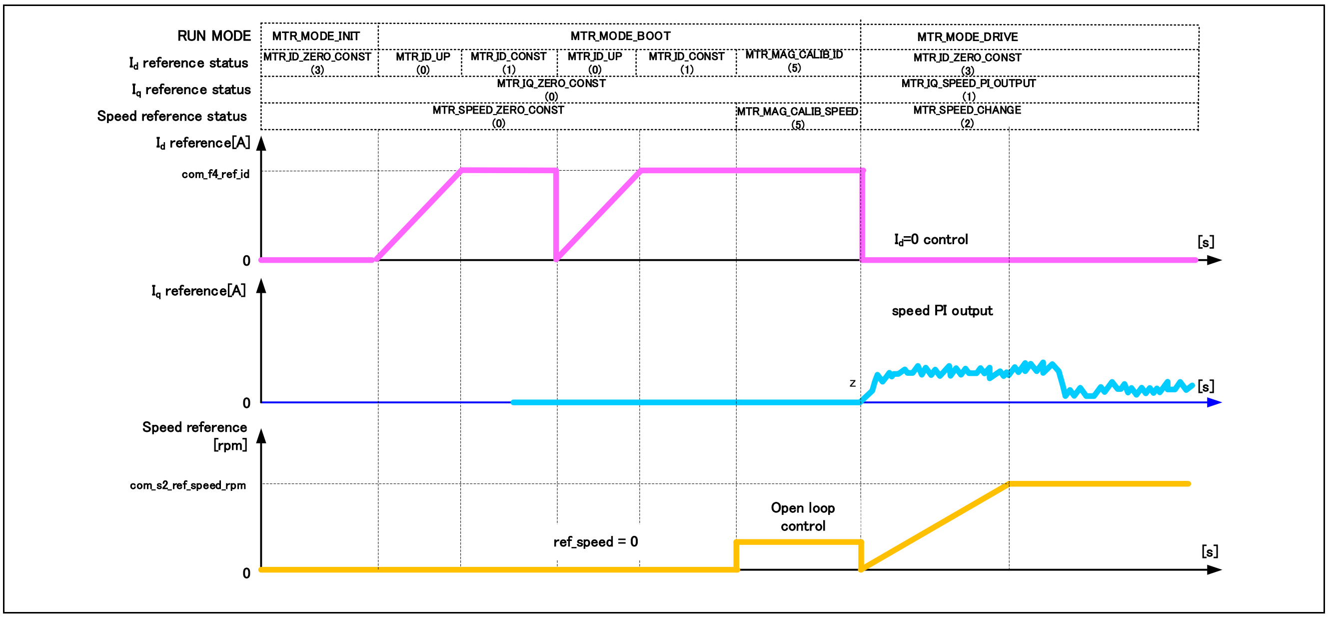


Figure 3-9 Startup Speed Control of Vector Control PMSM with analog Magnet sensor Software

Note: 1. Only on first start when using error correction.

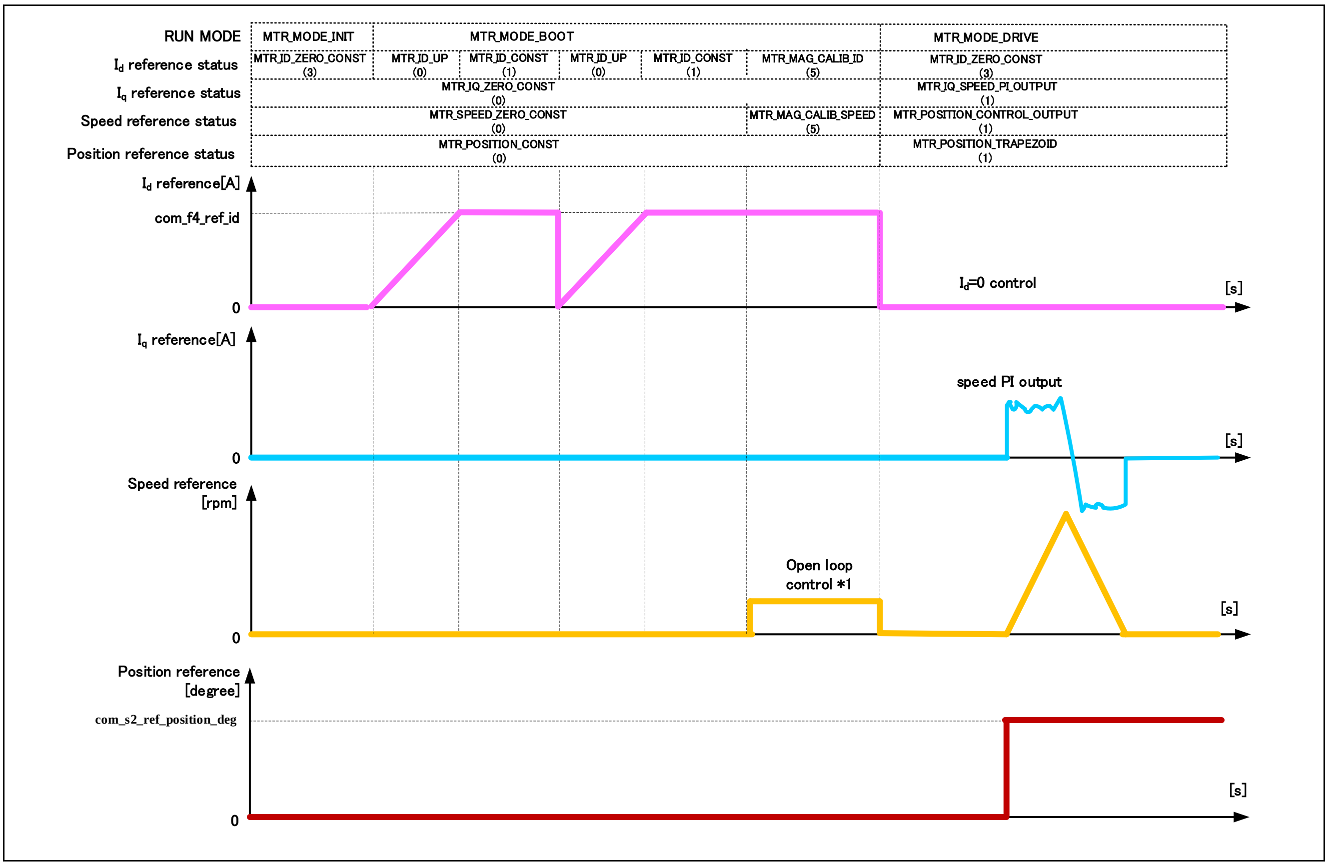


Figure 3-10 Startup Speed Control of Vector Control PMSM with analog Magnet sensor Software

3.1.8 System Protection Function

This control program 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 as well as 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. For hardware overcurrent value, refer to Renesas Solution Starter Kit 24V Motor Control Evaluation System for RX23T (Motor RSSK) Operation Manual.

  In addition, U, V, and W phase currents are monitored in over current monitoring cycle. When an over current is detected, the CPU executes emergency stop (software detection). The overcurrent limit value has rated current of motor and rated current of inverter board, and the lower one of them is set as the overcurrent limit value. In this environment, the rated current of the motor is used.

- Over voltage error

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

- Over-speed error

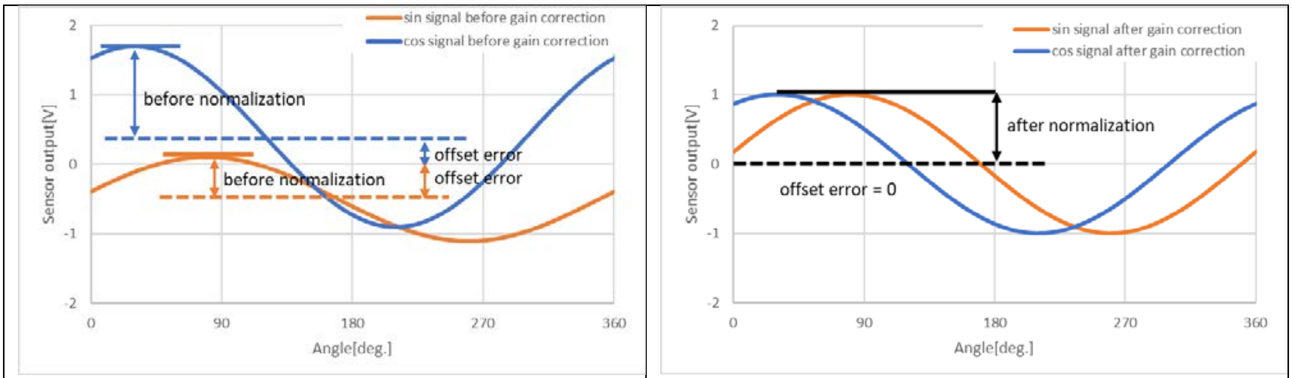
  The rotation speed is monitored in rotation speed monitoring cycle. The CPU performs emergency stop when the speed is over the limit value.

Table 3-5 Setting Values of the System Protection Function

Over-current error	Over-current limit value [A]	3.82
	Monitoring cycle [μs]	100
Over-voltage error	Over-voltage limit value [V]	28
	Monitoring cycle [μs]	100
Under-voltage error	Under-voltage limit value [V]	14
	Monitoring cycle [μs]	100
Over-speed error	Speed limit value [rpm]	1000
	Monitoring cycle [μs]	100

3.1.9 Error correction of analog magnetic sensor

This control program has a function to correct the analog output of the magnetic sensor. When an angle is detected from the sin signal and cos signal with an analog output sensor, the sensor output offset and output variation will cause an angle error. This control program can use the function to correct the sensor output. The concept of gain correction is shown in Figure 3-11 and the concept of phase correction is shown in Figure 3-12.

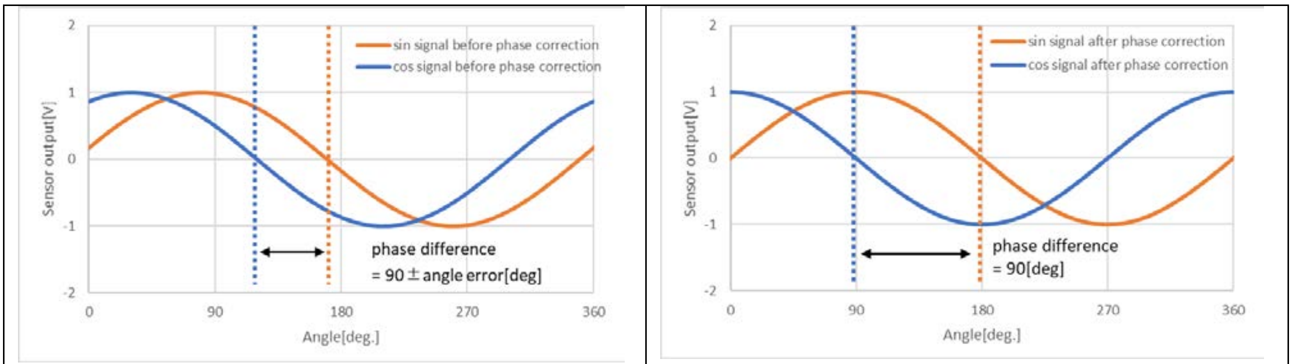


(a) Sensor signal before gain correction **(b) Sensor signal after gain correction**

Figure 3-11 Analog output error gain correction concept

When the sensor output correction function is enabled, the sensor output data acquisition processing is performed at the first start. It is used when calculating the correction coefficient with software based on the acquired data and calculating the angle from the sensor output. The output data of the sensor is acquired by open loop operation. Figure 3-12 shows the flow of data acquisition and correction.

In addition, the initial position information is acquired at the start because it is necessary to acquire the rotor position of the motor and the position offset of the detection magnet of the magnetic sensor regardless of whether the correction function is enabled or disabled.



(a) Sensor signal before phase correction **(b) Sensor signal after phase correction**

Figure 3-12 Analog output error phase correction concept

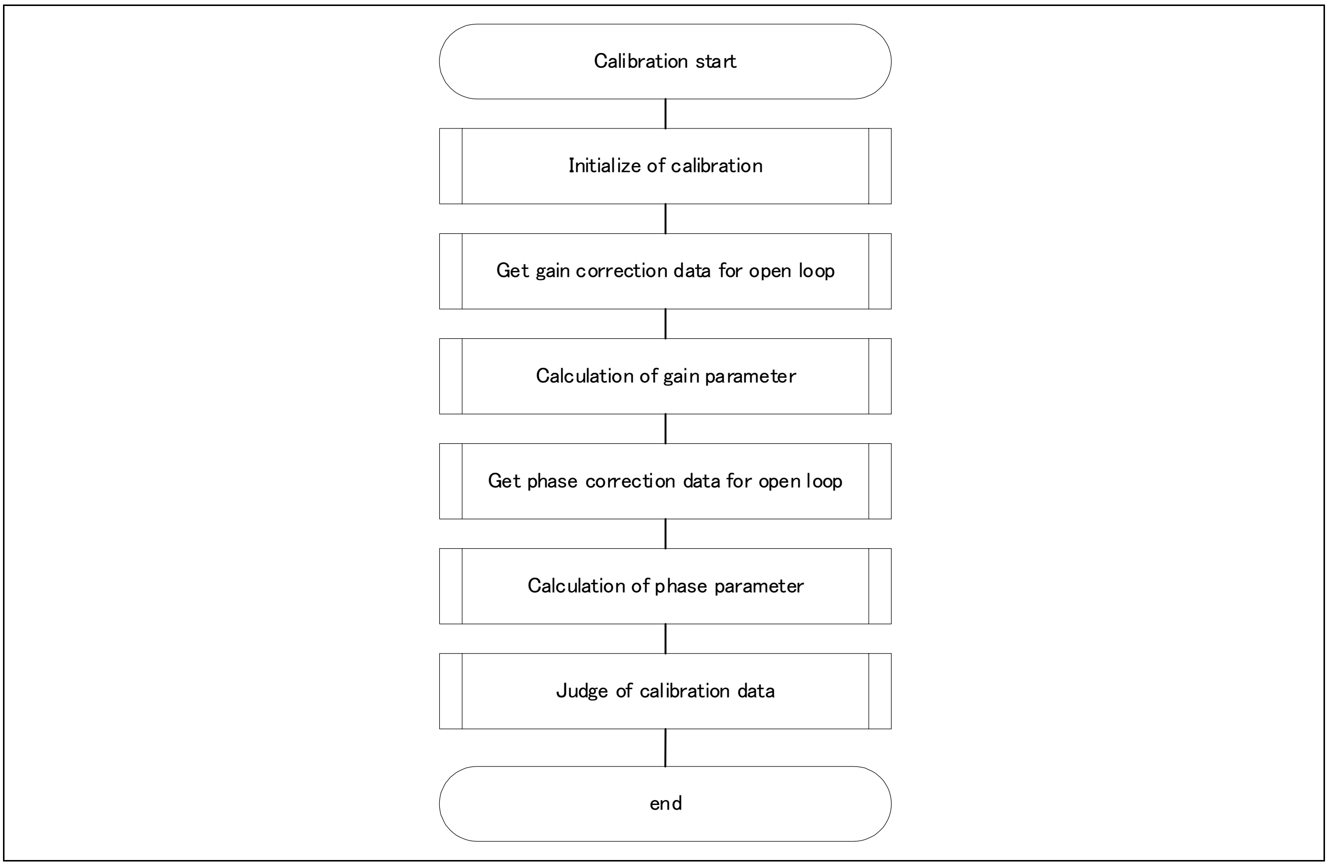


Figure 3-13 Error correction process flowchart

3.2 Function Specifications of Vector Control using magnet sensor Software

The control process of the target software of this application note is mainly consisted of 100[μs] period interrupt (carrier interrupt) and 1[ms] period interrupt. In the following Figure 3-14, the control process in the red part is executed every 100[μs] period, and the control process in the blue part is executed every 1[ms] period.

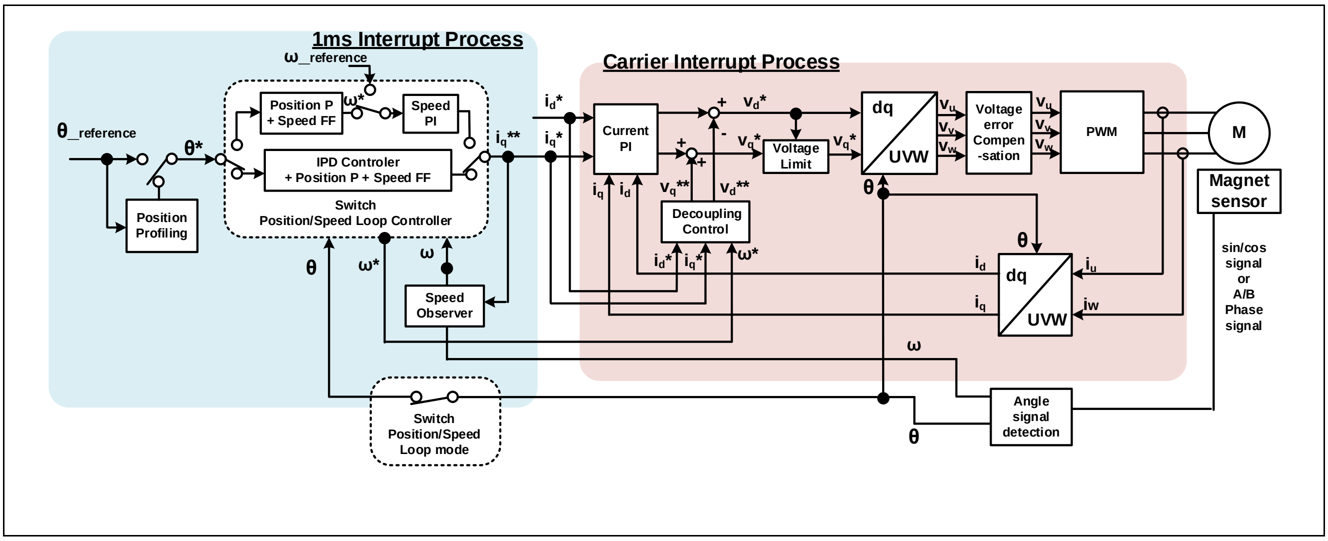


Figure 3-14 System Block of Vector Control with magnet sensor

This chapter shows the specification of 4 interrupt functions and functions executed in each interrupt cycle. In the following tables, only essential functions of the vector control are listed. Regarding the specification of functions not listed in following tables, refer to source codes.

Table 3-6 List of Control Functions 'mtr_interrupt.c'

File name	Function name	Process overview
r_mtr_interrupt_carrier.c	mtr_foc_carrier_interrupt Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: None	Calling every 100 [μs] • Current and voltage monitoring • Vector calculation • Current PI control • Position and speed calculation • PWM output setting
r_mtr_interrupt_1ms.c	mtr_foc_1ms_interrupt Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: None	Calling every 1 [ms] • Startup control • d-axis/q-axis current and speed reference set • Speed PI control

RENESAS

Table 3-7 List of Functions for 100μs interrupt [1/2]

File name	Function name	Process overview
r_mtr_ctrl_mrssk.c	mtr_get_current_iuiw Input: (float*) f4_iu_ad / U phase current A/D conversion value (float*) f4_iw_ad / W phase current A/D conversion value (uint8_t) u1_id / Motor ID Output: None	Obtaining the UVW 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.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	UVW phase current offset adjustment
	mtr_calib_current_offset Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: None	UVW phase current offset calculation
	mtr_mag_pos_speed_calc Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control (mtr_magnet_t *) st_mg / Structure pointer for magnet sensor Output: None	Position and speed calculation for analog magnet sensor signal
	mtr_angle_speed (mtr_mag_angle_speed) Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: None	Rotor phase and speed related process
	mtr_foc_voltage_limit Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: None	Voltage command value limit
r_mtr_foc_current.c	mtr_current_pi_control Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: None	Current PI
	mtr_foc_current_decoupling Input: (mtr_current_control_t *) st_cc / Current control structure (float) f4_speed_rad / speed (mtr_parameter_t *) mtr_para / motor parameter structure Output: None	Decupling control
	mtr_foc_voltage_limit Input: (mtr_foc_control_t *) st_foc / Structure pointer for vector control Output: None	Voltage command value
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 3-7 List of Functions for 100μs Interrupt [2/2]

File name	Function name	Process overview
r_mtr_volt_err_comp.c	mtr_volt_err_comp_main Input: (mtr_volt_comp_t *) st_volt_comp / Voltage error compensation structure (float*) p_f4_v_array / Three phase voltage compensation value array pointer (float*) p_f4_i_array / Three phase current compensation value array pointe (float) f4_vdc / Vdc value Output: None	Voltage error compensation
r_mtr_ctrl_rx24t.c	mtr_inv_set_uvw Input: (float) f4_modu / U phase modulation factor (float) f4_modv / V phase modulation factor (float) f4_modw / W phase modulation factor (uint8_t) u1_id / Motor ID Output: None	PWM output setting

Table 3-8 List of Functions for 1ms Interrupt

File name	Function name	Process overview
r_mtr_ctrl_rx24t.c	mtr_speed_calc_timer_start Input: (uint8_t) u1_id / Motor ID Output: None	Start for magnet sensor timer
r_mtr_ctrl_analog_magnet.c	mtr_mag_initial_pos_adjust Input: (mtr_magnet_t *) st_mg / Structure pointer for magnet sensor Output: None	Initialize position for analog magnet sensor
r_mtr_foc_control.c	mtr_set_pos_ref Input: (mtr_foc_control_t *) st_foc / FOC motor structure Output: (float32) f4_ref_pos_rad_calc / position command value	Setting the command value for position control
	mtr_set_speed_ref Input: (mtr_foc_control_t *) st_foc / FOC motor structure Output: (float32) f4_speed_ref_rad _calc / speed command value	Setting the command value for speed control
	mtr_set_iq_ref Input: (mtr_foc_control_t *) st_foc / FOC motor structure Output: (float32) f4_iq_ref_calc / q-axis current command value	Setting the q axis current command value
	mtr_set_id_ref Input: (mtr_foc_control_t *) st_foc / FOC motor structure Output: (float32) f4_id_ref_calc / d-axis current command value	Setting the d axis current command value

3.3 Macro Definitions of Vector Control Software Using Magnet sensor

Lists of macro definitions used in this control program are given from Table 3-9 to Table 3-12.

Table 3-9 List of Macro Definitions 'r_mtr_motor_parameter.h'

File name	Macro name	Definition value	Remarks
r_mtr_motor_parameter.h	MP_POLE_PAIRS	50	Number of pole pairs
	MP_MAGNETIC_FLUX	0.004574248f	Flux [Wb]
	MP_RESISTANCE	0.4202675f	Resistance [Ω]
	MP_D_INDUCTANCE	0.0009717637f	d-axis Inductance [H]
	MP_Q_INDUCTANCE	0.0009717637f	q-axis Inductance [H]
	MP_ROTOR_INERTIA	0.0000627562f	Rotor inertia [kgm^2]
	MP_NOMINAL_CURRENT_RMS	2.0f	Nominal torque [Arms]

Table 3-10 List of Macro Definitions 'r_mtr_control_parameter.h'

File name	Macro name	Definition value	Remarks
r_mtr_control_parameter.h	CP_POS_OMEGA	5	Natural frequency of the position loop [Hz]
	CP_SPEED_OMEGA	30	Natural frequency of the speed loop [Hz]
	CP_SPEED_ZETA	1.0f	Damping ratio of the speed loop
	CP_CURRENT_OMEGA	300f	Natural frequency of the current loop [Hz]
	CP_CURRENT_ZETA	1.0f	Damping ratio of the current loop
	CP_SOB_OMEGA	200	Natural frequency of the speed observer [Hz]
	CP_SOB_ZETA	1.0	Damping ratio of the speed observer
	CP_MIN_SPEED_RPM	0	Minimum speed (mechanical) [rpm]
	CP_MAX_SPEED_RPM	500	Maximum speed (mechanical) [rpm]
	CP_SPEED_LIMIT_RPM	1000	Limit speed (mechanical) [rpm]
	CP_OL_IP_REF	1.5f	d-axis current command value [A]

Table 3-11 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	Deadtime [μs]
	IP_CURRENT_RANGE	20.0f	current sensing range (-10 [A] to 10 [A])
	IP_VDC_RANGE	111.0f	voltage sensing range (0 [V] to 111 [V])
	IP_INPUT_V	24.0f	input DC voltage [V]
	IP_CURRENT_LIMIT	5.0f	Current limit[A]*
	IP_OVERVOLTAGE_LIMIT	28.0f	Over voltage limit [V]
	IP_UNDERVOLTAGE_LIMIT	14.0f	Under voltage limit [V]

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

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

File name	Macro name	Definition value	Remarks
config.h	IP_MRSSK	—	Inverter select macro
	RX24T_MRSSK	—	MCU select macro
	MP_iSS57_20	—	Motor select macro
	CP_iSS57_20	—	
	CONFIG_DEFAULT_UI	ICS_UI	Select default UI ICS_UI: Use the Analyzer for RMW BOARD_UI: Use board interface
	USE_VOLT_ERR_COMP	1	Voltage error compensation 0: Disable 1: Enable
	POS_CTRL_MODE	MTR_CTRL_IPD	Select position control mode MTR_CTRL_PID: PID controller MTR_CTRL_IPD: IPD controller
	LOOP_MODE	MTR_LOOP_POSITION	Select control loop mode MTR_LOOP_SPEED: speed loop mode MTR_LOOP_POSITION: position loop mode
	GAIN_MODE	MTR_GAIN_DESIGN_MODE	Gain mode MTR_GAIN_DESIGN_MODE: 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
	USE_CALIB	0	Error calibration select MAG_CALIB_INACTIVE: Calibration inactive MAG_CALIB_ACTIVE: Calibration active

3.4 Control Flowcharts

Detailed processing of the system block diagram of Figure 3-15 is shown in Figures 3- 15 to Figure 3-19.

3.4.1 Main Process

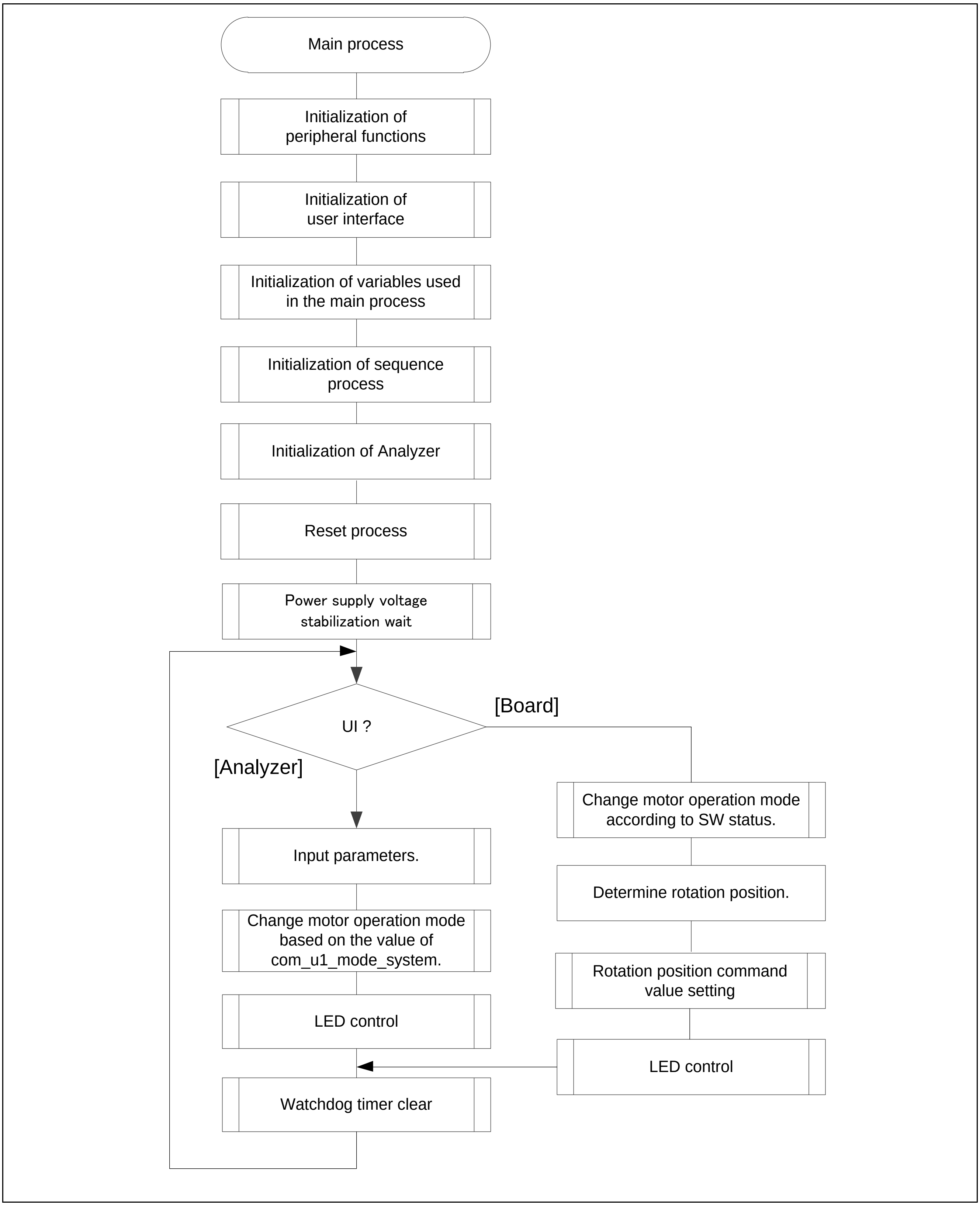


Figure 3-15 Main Process Flowchart

3.4.2 100 [us] Interrupt Handling

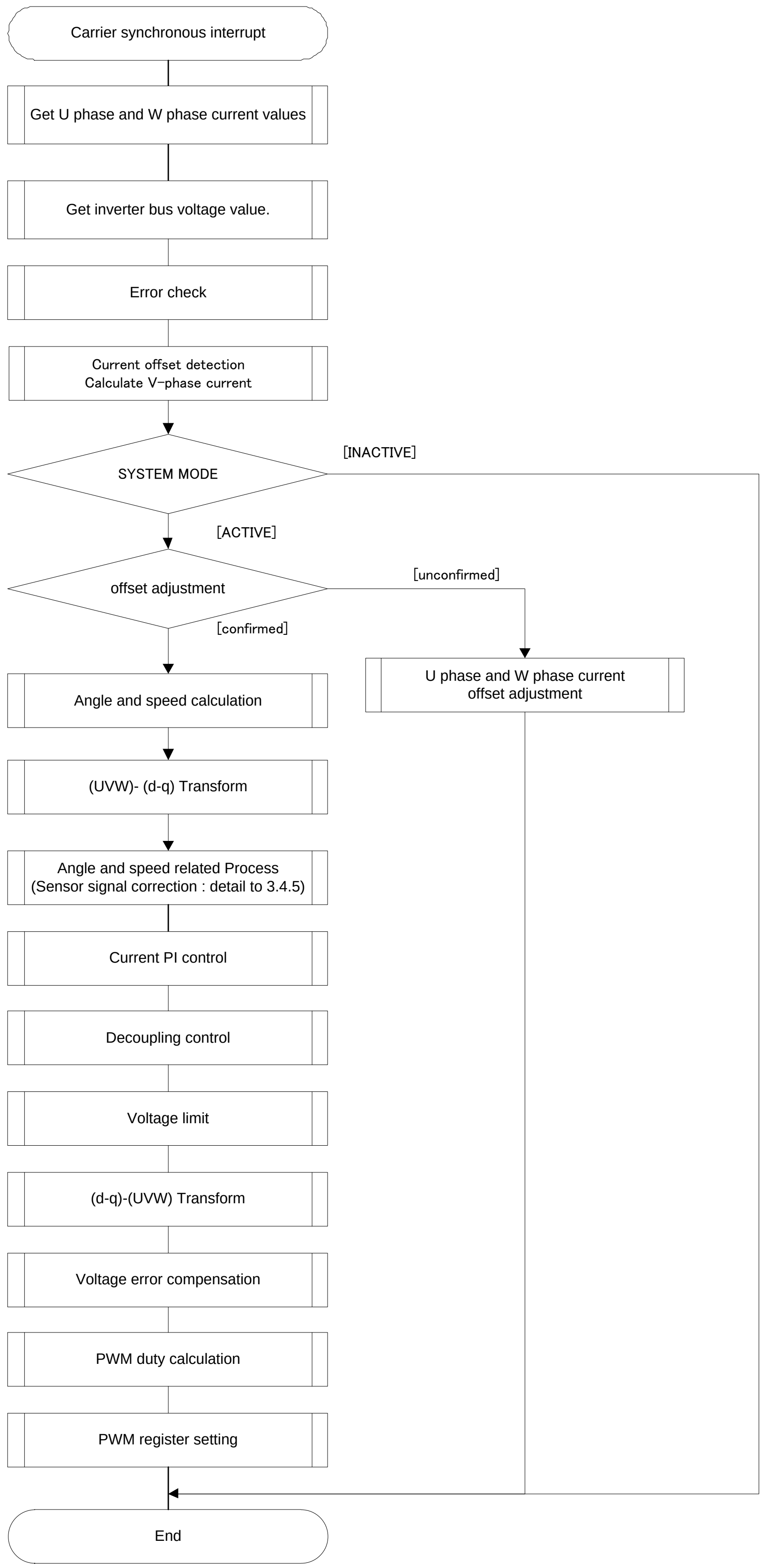


Figure 3-16 100 [µs] Cycle Interrupt Handling

3.4.3 1 [ms] Interrupt Handling

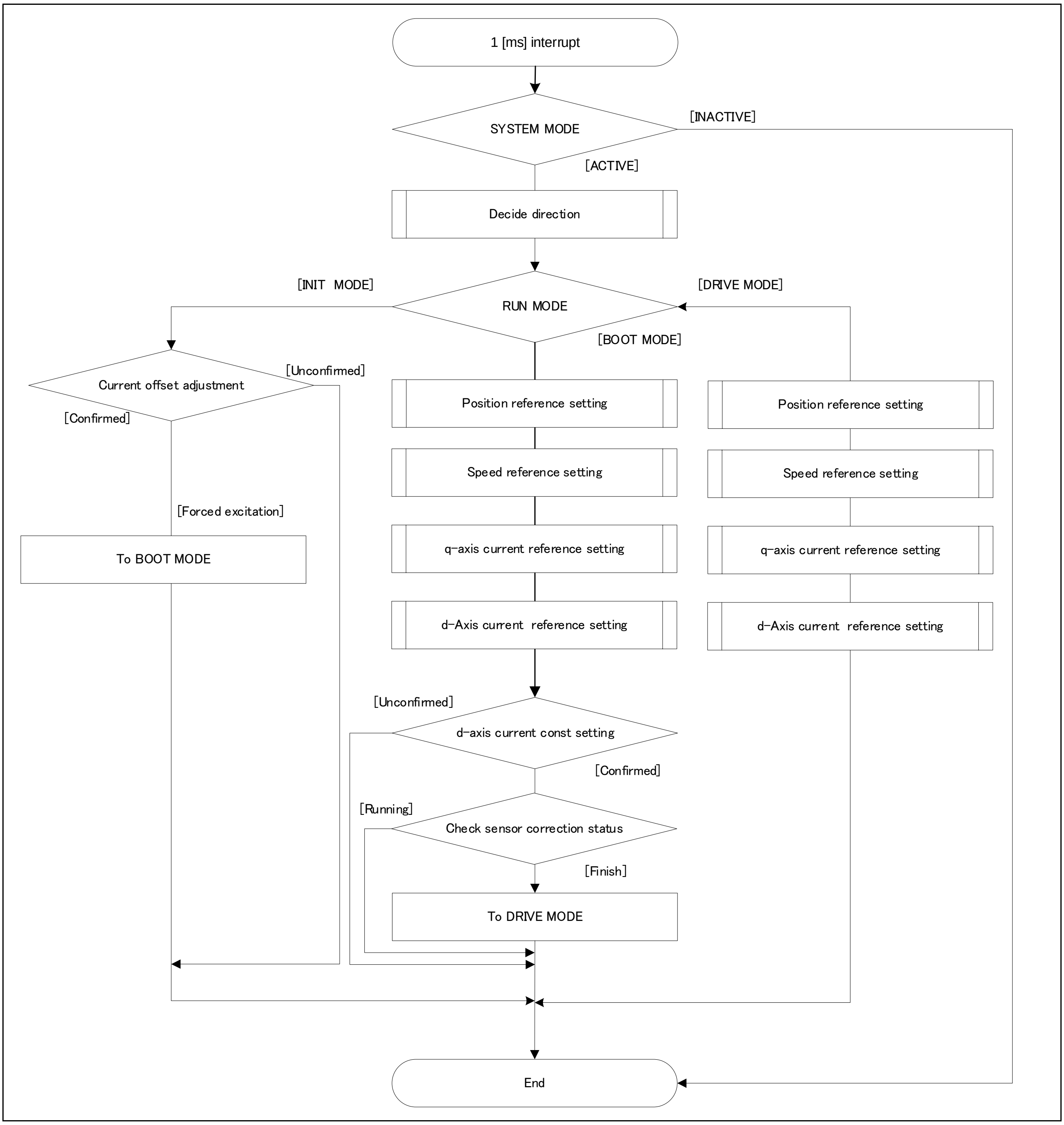


Figure 3-17 1 [ms] Interrupt Handling

3.4.4 Over Current Detection Interrupt Handling

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.

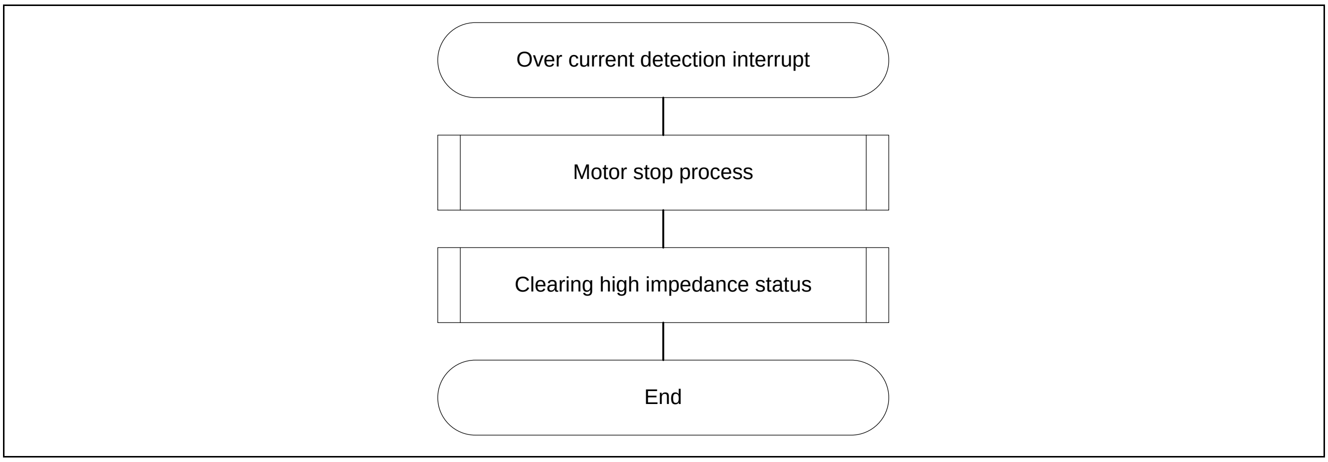


Figure 3-18 Over Current Detection Interrupt Handling

3.4.5 Angle and speed related process

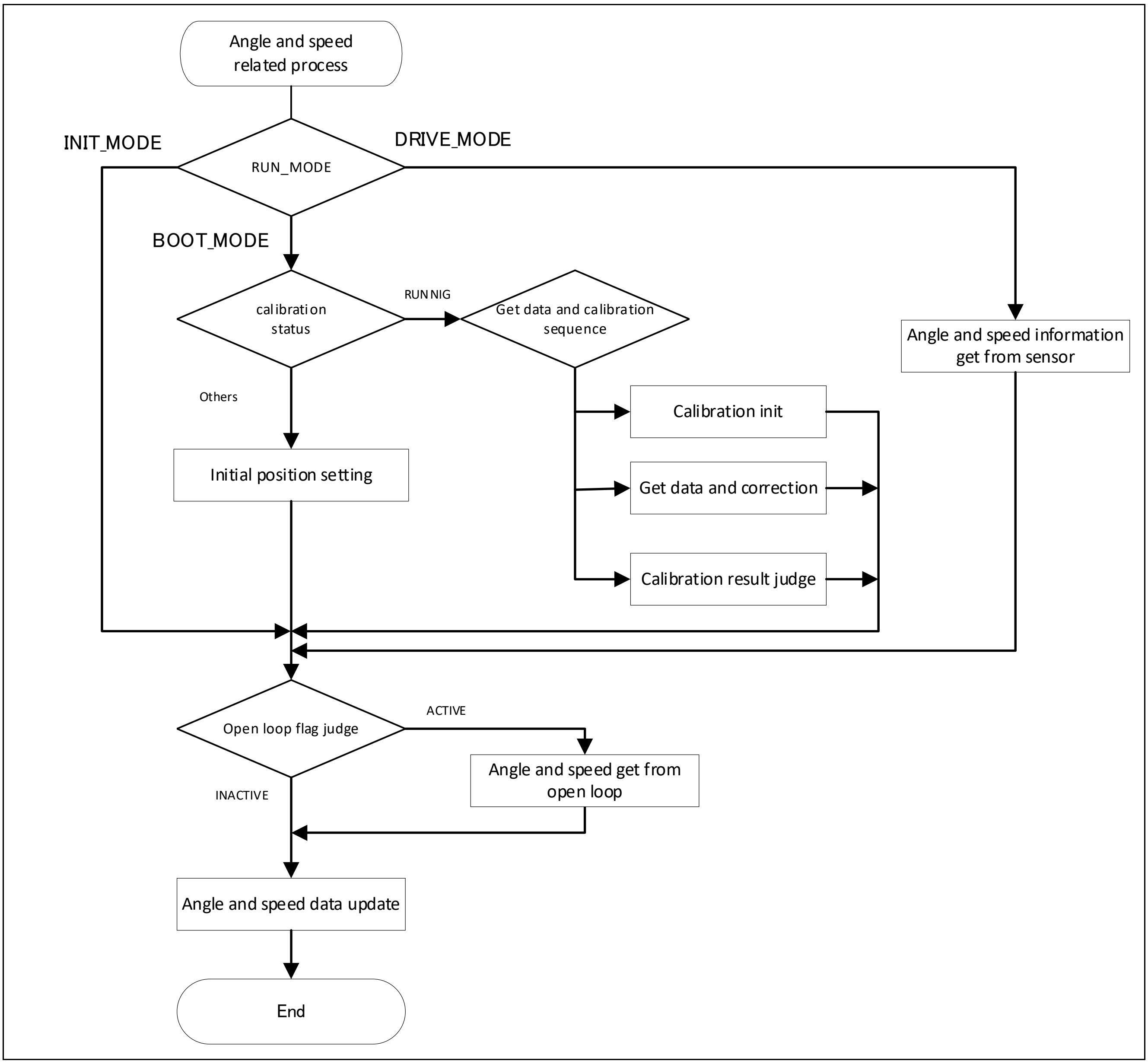


Figure 3-19 Detail of angle and speed related process

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 V.2.00 User's Manual' for usage and more details. 'Renesas Motor Workbench' can be downloaded from Renesas Electronics Corporation website.

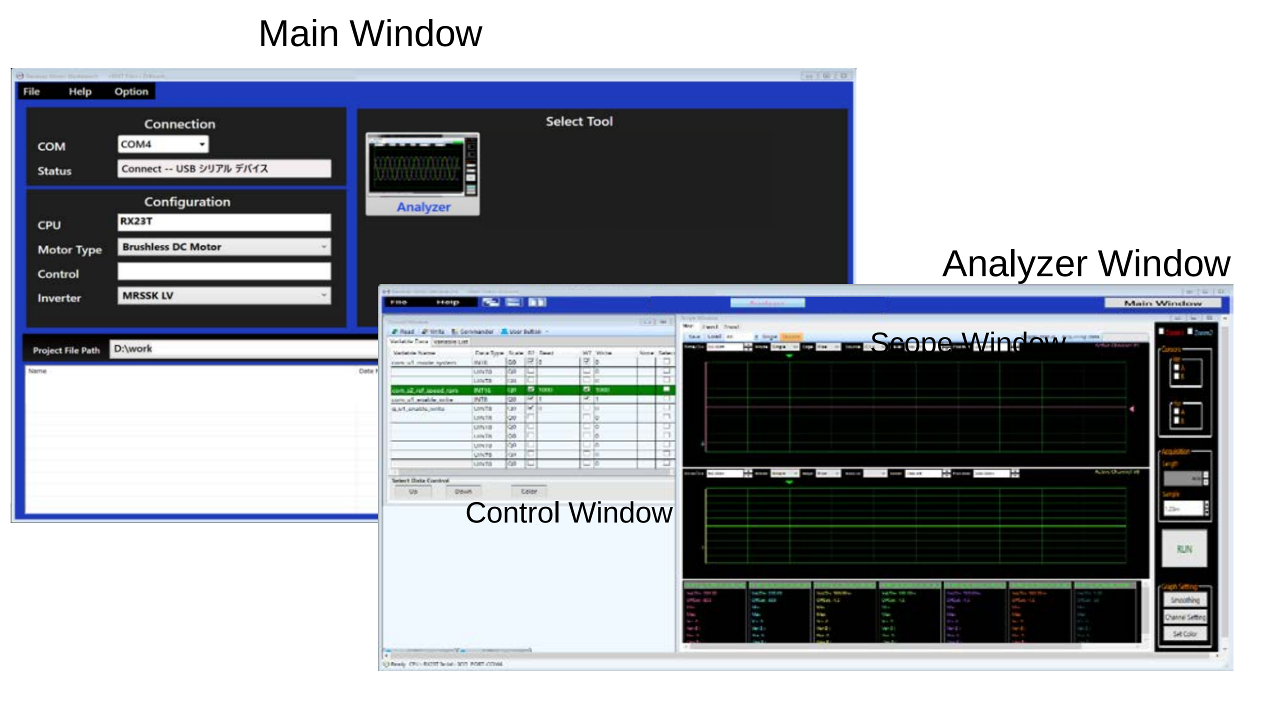
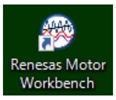


Figure 4-1 Renesas Motor Workbench – Appearance

<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.
(3) Select the RMT file from following location of e2studio/CS+ project folder.
'[Project Folder]/ application/user_interface/ics/'
(4) Use the 'Connection' [COM] select menu to choose the COM port.
(5) Click on the 'Analyzer' icon of Select Tool panel to open Analyzer function window.
(6) Please refer to '4.3 Operation Example for Analyzer' for motor driving operation.

4.2 List of Variables for Scope Function ‘Analyzer’

Table 4-1 is a list of variables for Analyzer. These variable values are reflected to the protect variables when the same values as of g_u1_enable_write are 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_u1_ctrl_loop_mode	uint8_t	Control loop mode switch 0: Speed control 1: Position control (default)
com_u1_ctrl_method_mode	uint8_t	Control method switch 0: PID control (Position P/Speed PI/Current PI) 1: IPD control (position•Speed IPD + Position FF + Speed FF + Position P/ Current PI) (default) FF: Feed-forward control
com_u1_position_input_mode	uint8_t	Position reference input mode switch 0: 0 output 1: direct input 2: Position profiling (default)
com_s2_ref_position_deg	int16_t	Position command value (mechanical) [degree]
com_s2_ref_speed_rpm	int16_t	Speed command value (mechanical) [rpm]
com_u2_min_speed_rpm	uint16_t	Minimum speed (mechanical) [[rpm]
com_u2_max_speed_rpm	uint16_t	Maximum speed (mechanical) [rpm]
com_u2_speed_limit_rpm	uint16_t	Overspeed Limit (mechanical) [rpm]
com_u2_pos_interval_time	uint16_t	Time interval of the position command changes [s]
com_u2_pos_dead_band	uint16_t	Dead band of position
com_u2_pos_band_limit	uint16_t	Positioning complete range
com_u2_offset_calc_time	uint16_t	Calculation time of current offset value [ms]
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_f4_nominal_current_rms	float	Nominal current [Arms]

Table 4-1 List of Variables for Analyzer (2/2)

Variable name	Type	Content
com_f4_current_omega	float	Natural frequency of the current loop [Hz]
com_f4_current_zeta	float	Damping ratio of the current loop
com_f4_speed_omega	float	Natural frequency of the speed loop [Hz]
com_f4_speed_zeta	float	Damping ratio of the speed loop
com_f4_pos_omega	float	Natural frequency of the position loop [Hz]
com_f4_sob_omega	float	Natural frequency of the speed observer [Hz]
com_f4_sob_zeta	float	Damping ratio of the speed observer
com_f4_id_kp	float	d axis current PI control proportional term gain
com_f4_id_ki	float	d axis current PI control integral term gain
com_f4_iq_kp	float	q axis current PI control proportional term gain
com_f4_iq_ki	float	q axis current PI control integral term gain
com_f4_speed_kp	float	Speed PI control proportional term gain
com_f4_speed_ki	float	Speed PI control integral term gain
com_f4_pos_kp	float	Position control proportional term gain (PID) proportional term gain+ position control proportional term gain (IPD)
com_f4_ipd_speed_k_ratio	float	Speed control gain ratio for IPD
com_f4_ipd_pos_kp_ratio	float	Position control proportional term gain ratio for IPD
com_f4_ipd_err_limit_1	float	Position error limit for IPD
com_f4_ipd_err_limit_2	float	Position error limit for IPD
com_f4_accel_time	float	Acceleration time [s] (for position control)
com_f4_id_ref_open	float	d-axis current command value [A]
com_f4_id_up_time	float	d-axis current command value addition time [ms]
com_f4_limit_speed_change	float	Acceleration limit [s] (for speed control)
com_u1_enable_write	uint8_t	Enabled to rewriting variables

The primary variables that are frequently observed during the motor driving evaluation are listed in Table 4-2 Please refer when using Analyzer function. Regarding variables not listed in Table 4-2, refer to source codes.

Table 4-2 List of Primary Variables for Magnet sensor Vector Control

Name of primary variable for magnet sensor Vector Control	Type	Content
st_foc.u2_error_status	uint16_t	error status
st_foc.st_cc.f4_id_ref	float	d-axis current command value [A]
st_foc.st_cc.f4_id_ad	float	d-axis current [A]
st_foc.st_cc.f4_iq_ref	float	q-axis current command value [A]
st_foc.st_cc.f4_iq_ad	float	q-axis current [A]
st_foc.f4_iu_ad	float	W phase current A/D conversion value [A]
st_foc.f4_iv_ad	float	V phase current A/D conversion value [A]
st_foc.f4_iw_ad	float	W phase current A/D conversion value [A]
st_foc.st_cc.f4_vd_ref	float	d-axis output voltage command value [V]
st_foc.st_cc.f4_vq_ref	float	q-axis output voltage command value [V]
st_foc.f4_refu	float	U phase voltage command value [V]
st_foc.f4_refv	float	V phase voltage command value [V]
st_foc.f4_refw	float	W phase voltage command value [V]
st_foc.st_sc.f4_ref_speed_rad_ctrl	float	Command value for speed control (Electrical) [rad/s]
st_foc.st_sc.f4_speed_rad	float	Speed (Electrical) [rad/s]
st_foc.st_pc.f4_ref_pos_rad_ctrl	float	Command value for Position control (Electrical) [rad]
st_foc.st_pc.f4_pos_rad	float	Position (Electrical) [rad]

4.3 Operation Example for Analyzer

This section shows an example below for motor driving operation using Analyzer. Operation is using 'Control Window' of the analyzer. Regarding specification of 'Control Window', refer to 'Renesas Motor Workbench V.2.00 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'
  - 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.

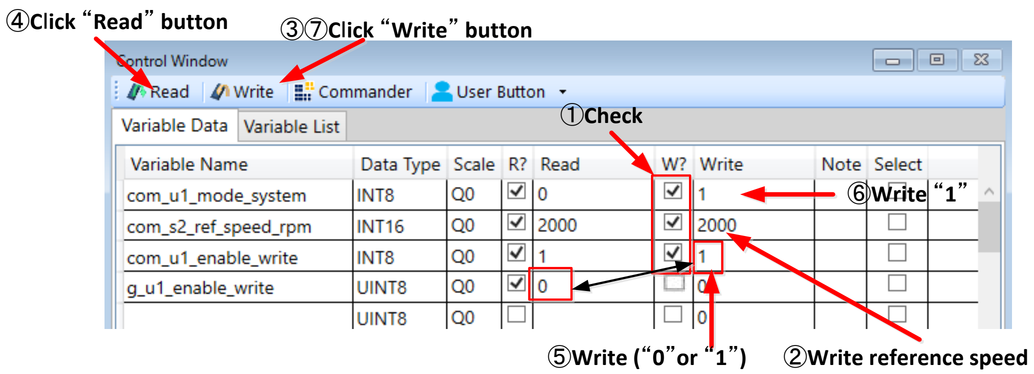


Figure 4-2 Procedure – Driving the Motor

- Stop the motor
  - ① Write '0' in the [Write] box of 'com_u1_mode_system'
  - ② Click the 'Write' button.

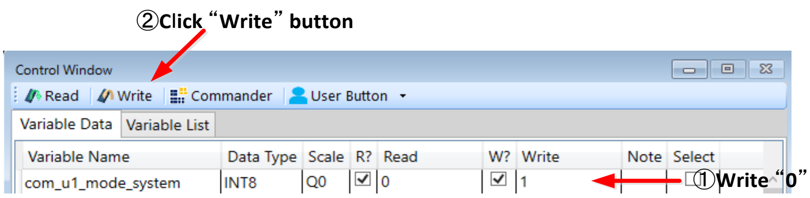


Figure 4-3 Procedure – Stop the Motor

- Error cancel operation
  ① Write ‘3’ in the [Write] box of ‘com_u1_mode_system’
  ② Click the ‘Write’ button.

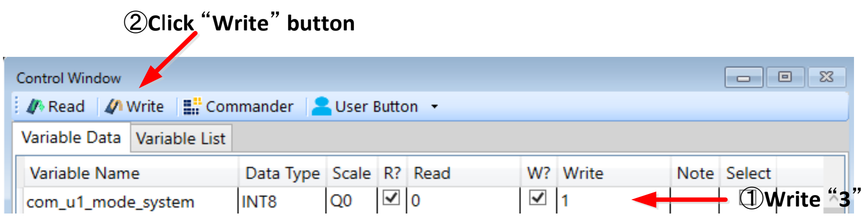


Figure 4-4 Procedure – Error Cancel Operation

4.4 Operation Example for User Button

This section shows an example of a motor driving operation using User Button.

- Driving or Stop the motor
  By setting as shown in Figure 4-5, driving and stopping changes each time the button is pressed.

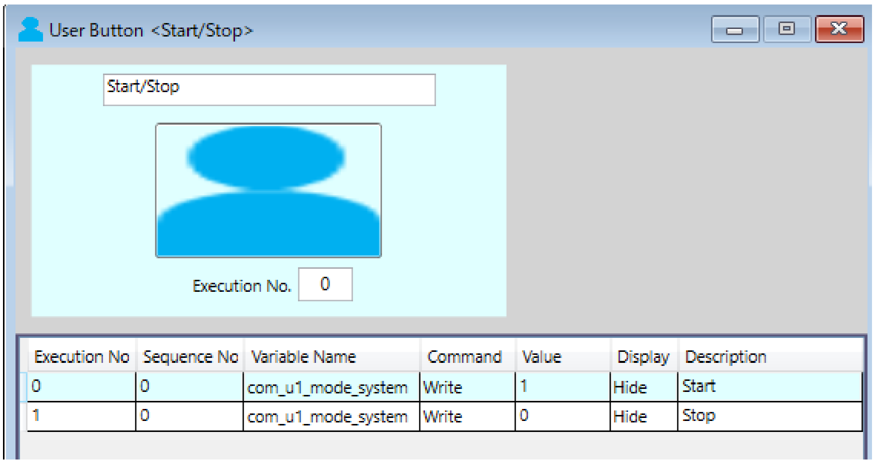


Figure 4-5 Driving or Stop the Motor

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

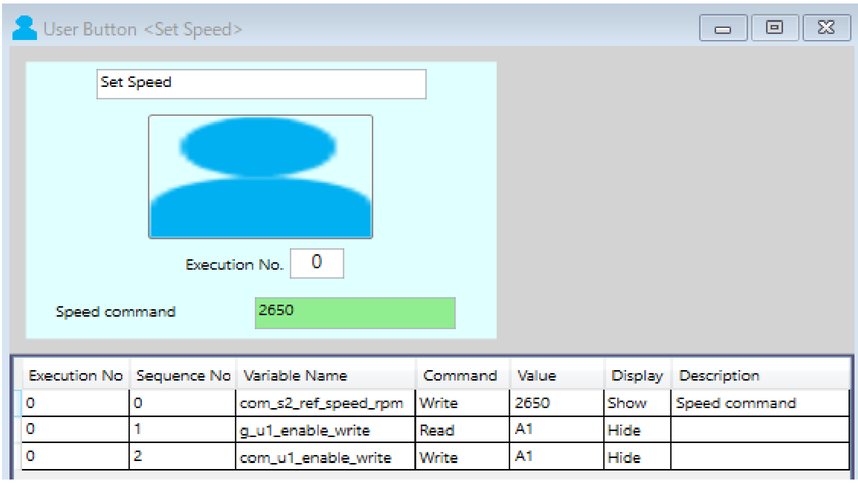


Figure 4-6 Change speed

5. Test results

The waveforms during control are shown as the test results using each sensor.

5.1 Speed control with TAD2141

■Measurement condition: The motor rotates at a constant speed of 100 rpm. See Table 3-11 for various control parameters.

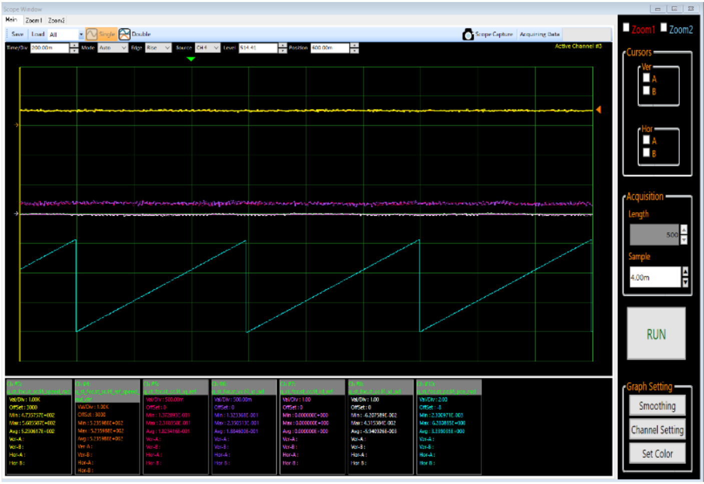


Figure 5-1 Speed control using TAD2141

Yellow: Speed [rad/s], Orange: Reference speed [rad/s]

Red: q-axis current reference value [A], Purple: q-axis current value [A], Pink: d-axis current reference value [A], White: d-axis current value [A], Light blue: Angle information calculated from the sensor (mechanical angle) [rad]

5.2 Position control with TAD2141

■Measurement condition: The motor rotates 2 times in the CW direction. Following control parameters were used: Natural frequency of the position loop 10[Hz],Natural frequency of the speed loop 30[Hz], Damping ratio of the speed loop 1.0f, Natural frequency of the current loop 400[Hz], Damping ratio of the current loop 1.0f.

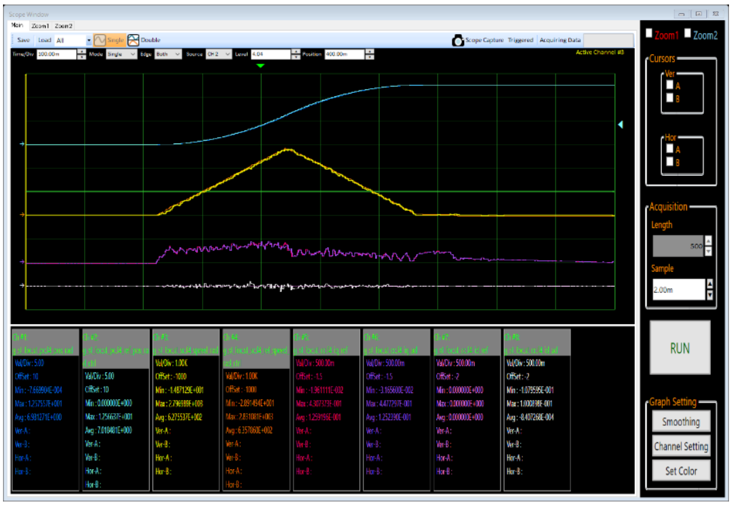


Figure 5-2 Position control using TAD2141

Blue: Angle calculated from sensor (mechanical) [rad], Light blue: Rotation angle reference value [rad]

Yellow: Detection speed [rad/s], Orange: Reference speed [rad/s]

Red 2: q-axis current reference value [A], Purple: q-axis current value [A], Pink: d-axis current reference value [A], White: d-axis current value [A]

5.3 Speed control with TAS2143

■Measurement condition: The motor rotates at a constant speed of 100 rpm. See Table 3-11 for various control parameters.

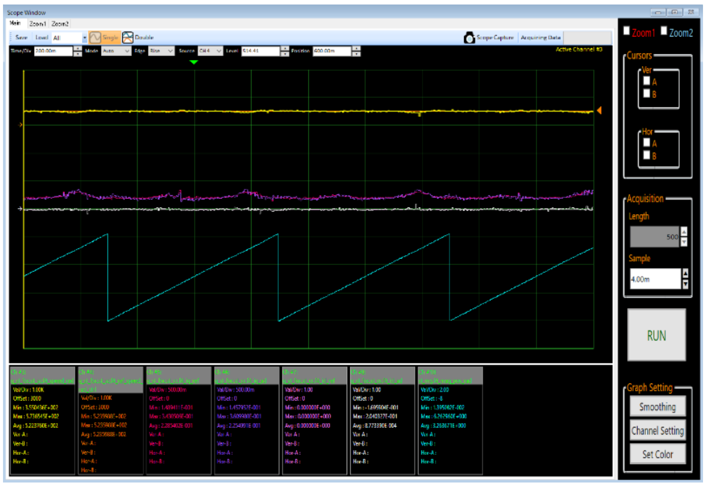


Figure 5-3 Speed control using TAS2143

Yellow: Speed [rad/s], Orange: Reference speed [rad/s]

Red: q-axis current reference value [A], Purple: q-axis current value [A], Pink: d-axis current reference value [A], White: d-axis current value [A]

Light blue: Angle information calculated from the sensor (mechanical angle) [rad]

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	Sep.16. 20	—	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

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/.