RL78/F14

R01AN3797EJ0100
Rev.1.00
Mar 31, 2017

Motor control by RL78/F14 micro controller sensorless vector control of permanent magnetic synchronous motor

Summary

This application note aims at explaining the sample program for operating the sensorless vector control of permanent magnetic synchronous motor, by using functions of RL78/F14.

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

Operation checking device

Operations of the sample program are checked by using the following device.

• RL78/F14(R5F10PLJ)

Contents

1. Overview

This application note explains the sample program of the sensorless vector control of permanent magnetic Synchronous motor (henceforth referred to as PMSM) by using the RL78/F14 micro controller.

1.1 Usage of the system

This system (sample program) enables sensorless vector control by using "ECU001-F14-12V" note 1 and BLDC motor "BLY171S-15V-8000" note 2 for motor control.

For installation and technical support of "ECU001-F14-12V", contact Sales representatives and dealers of Renesas Electronics Corporation.

Notes: 1. Evaluation board "ECU001-F14-12V" is products of Desk Top Lab Inc.
(http://www.desktoplab.co.jp/)
2. BLDC Motor "BLY171S-15V-8000" is products of Anaheim Automation Inc..
(http://www.anaheimautomation.com/)

1.2 Development environment

(1) Software development environment

Integrated development environment	CS+ for CA,CX (V3.03.00)
	CS+ for CC (V4.01.00)
	IAR Embedded Workbench (Ver. 7.4.1.4269)

(2) Hardware environment

On-chip debug emulator	E1
Micro controller used	RL78/F14(R5F10PLJ)
Compiler	CA78K0R(Ver.V1.7.2) CC-RL(Ver. 1.03.00.00) EWRL78 (Ver. 2.21.1)
Compiler Option	Normal
Inverter board for motor control	ECU001-F14-12V
BLDC Motor	PMSM(BLY171S-15V-8000)

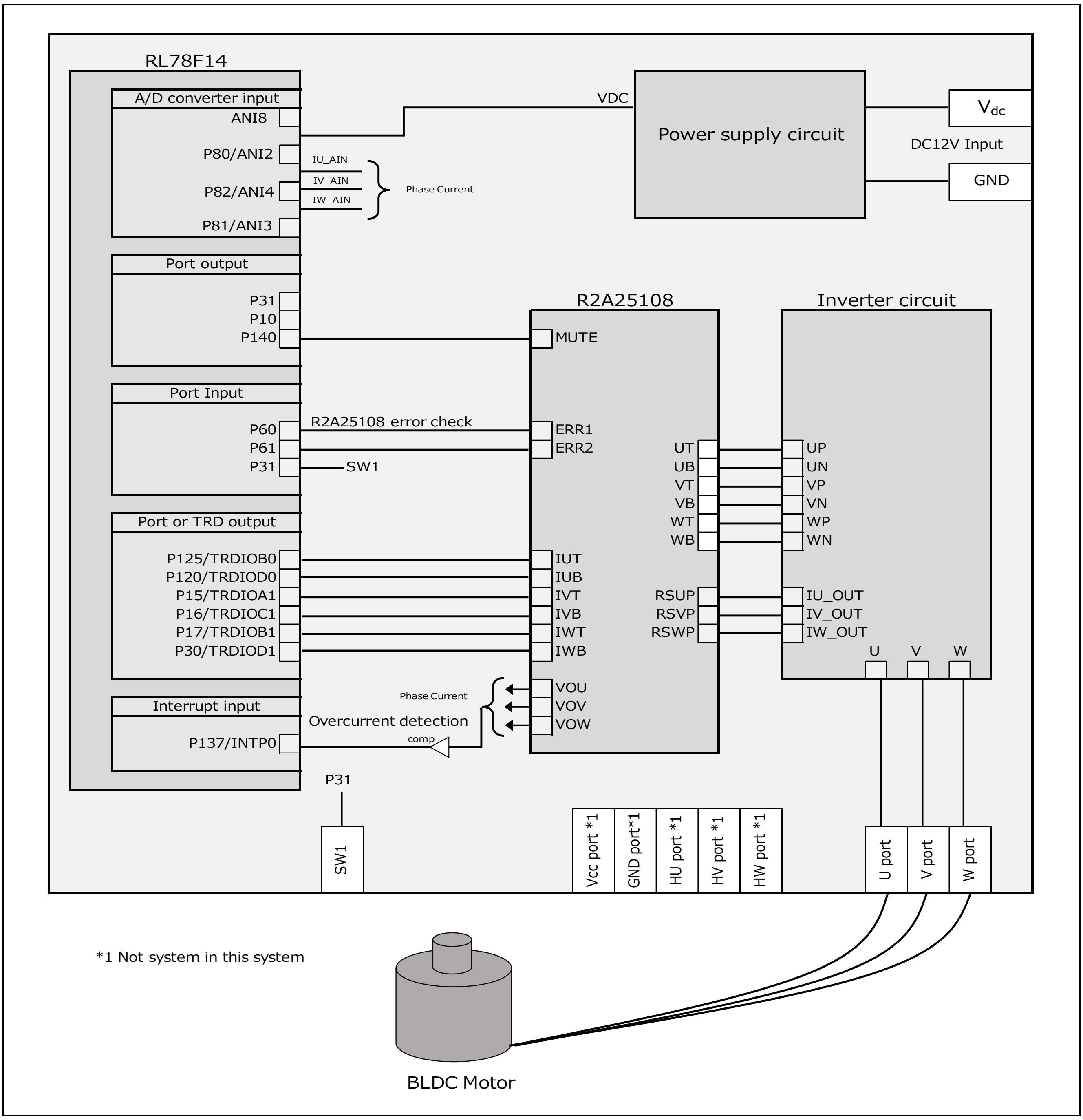
2. System overview

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 port interfaces of RL78/F14 micro controller of this system is given in Table2-1

Table 2-1 Port interfaces

Port name	Function
P86 / ANI8	Inverter bus voltage measurement
P80 / ANI2	U phase current measurement
P82 / ANI4	V phase current measurement
P81 / ANI3	W phase current measurement
P125 / TRDIOB0	Complementary PWM output (Up)
P15 / TRDIOA1	Complementary PWM output (Vp)
P17 / TRDIOB1	Complementary PWM output (Wp)
P120 / TRDIOD0	Complementary PWM output (Un)
P16 / TRDIOC1	Complementary PWM output (Vn)
P30 / TRDIOD1	Complementary PWM output (Wn)
P140	Pre-Driver MUTE output
P31	SW1 input
P60	ERR1 error detection input
P61	ERR2 error detection input
P137 / INTP0	Over current detect circuit input
RESET	RESET

2.2.2 Peripheral functions

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

Table 2-2 List of the Peripheral Functions

Peripheral function	Usage
10bits A/D converter (ANI2, ANI3, ANI4, ANI8)	・U、V、W phase current measurement ・Inverter bus voltage measurement
Timer Array Unit (TAU)	・1 [ms] interval timer
Timer RD (TRD)	・Complementary PWM output (six outputs)
Input INTP0	In the case of over current detection, set PWM output to high impedance

(1) 10-bit A/D converter

U phase current (Iu), V phase current (Iv), W phase current (Iw), and inverter bus voltage (Vdc) are measured by using ‘10-bit A/D converter’.

For A/D conversion, set the channel selection mode to ‘Select mode’ and the conversion operation mode to ‘One shot conversion mode’ (use software trigger).

(2) Timer Array Unit (TAU)

1 [ms] interval timer uses the channel 1 of the Timer Array Unit (TAU).

(3) Timer RD (TRD)

The 6-phase PWM output with dead time is performed by using the complementary PWM mode.

(4)Input INTP0

Set PWM output to high impedance

2.3 Software configuration

2.3.1 Software configuration

Folder and file configuration of the sample program is given below.

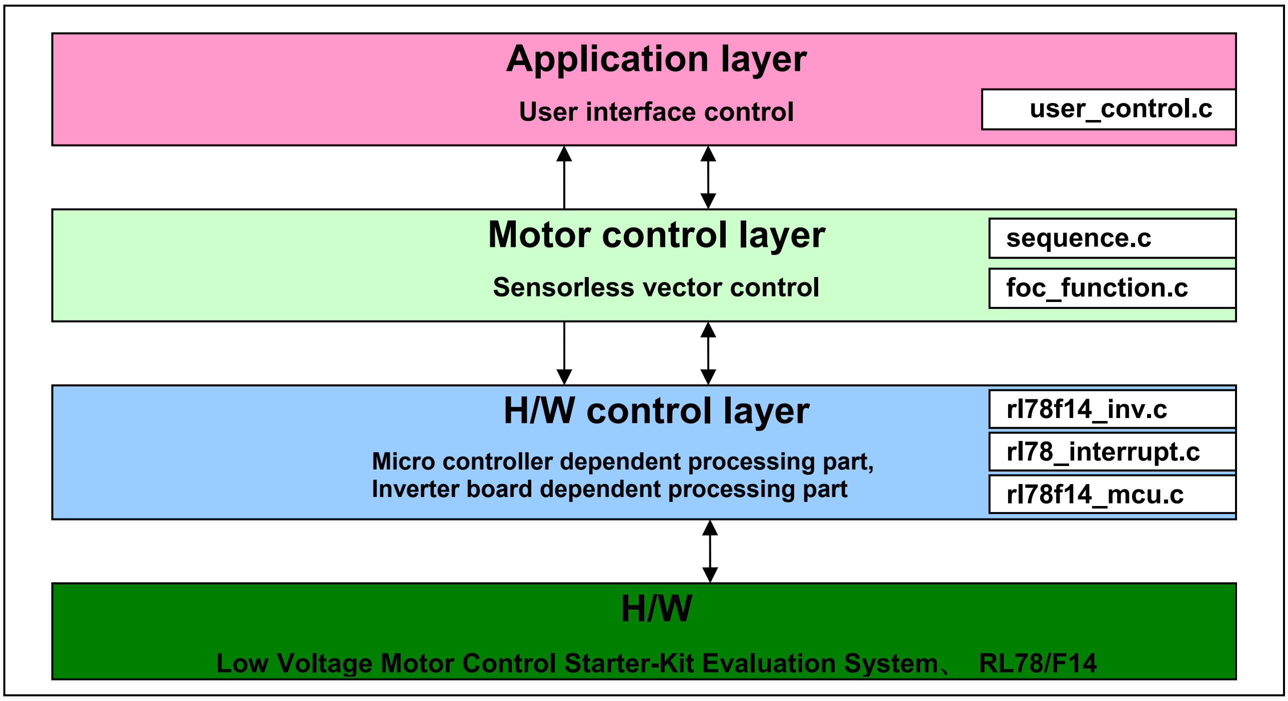
Table 2-4 Folder and File Configuration of the Sample Program

RL78F14_SSL_FOC180_V0100	inc	control_parameter.h	Control Parameter Header
		foc_function.h	Sensor-less vector control Header
		motor_parameter.h	Motor Parameter Header
		rl78_common.h	Common Header
		rl78_interrupt.h	Interrupt Header
		rl78f14_inv.h	Inverter Header
		rl78f14_mcu.h	MCU resister setting
		rl78f14_system.h	F14 System Header
		sequence.h	Sequence Header
		user_control.h	User Header
	lib	rl78_foclib.lib	Sensor-less vector calculation library
		R78_scale_ctrl.h	Adjustment scaling header
		rl78_scaling_math.lib	Calculation library
	src	foc_function.c	Sensor-less vector control
		main.c	Main function, User interface
		rl78_interrupt.c	Interrupt Pandora
		rl78f14_inv.c	Inverter control
		rl78f14_mcu.c	MCU resister setting
		sequence.c	Sequence control
		user_control.c	User control

2.3.2 Module configuration

Module configuration of the sample program is described below.

Figure 2-2 Module Configuration of the Sample Program



2.3.3 Software specification

Basic specifications of the software are given in Table 2-5.

Table 2-5 Basic Specifications of the Software

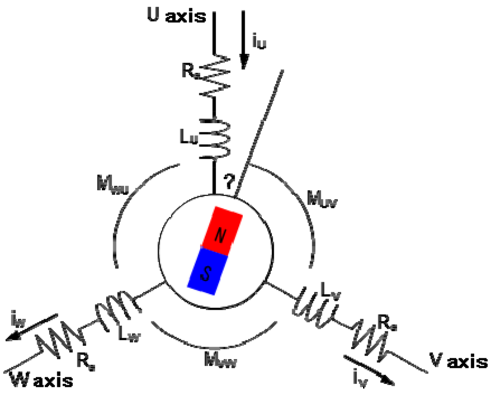
Item	Content
Control method	Sensor-less Vector control
Motor rotation start/stop	- Start by SW1 push down short time. Speed can change 0rpm->800rpm→4000rpm->6000rpm - Stop by SW1 push down long time or driver error detection
Position detection of rotor magnetic pole	Sensorless
Carrier frequency (PWM)	16 [kHz]
Control cycle	187.5[μs] (carrier cycle ×3)
Rotation speed control range	CW/CCW : 800 [rpm] ～ 6000 [rpm] initial value 0[rpm] Speed control value change with SW1
Processing stop for protection	・Disables the motor control signal output (six outputs), under any of the following four conditions. 1. Current of each phase exceeds 10 [A] (monitored per 187.5 [μs]) 2. Inverter bus voltage exceeds 28 [V] (monitored per 187.5[μs]) 3. Inverter bus voltage is less than 6 [V] (monitored per 187.5 [μs]) 4. Rotation speed exceeds 6600 [rpm] (mechanical angle) (monitored per 187.5 [μs]) ・In the case of over current detection, set the PWM output to high impedance (“Low” is input to the INTP0 port).

3. Motor control method

The SPMSM vector control used in the sample program is explained here.

3.1 Voltage equation of the motor control system

Voltage equation of the permanent magnetic synchronous motor (Figure 3-1) having the magnetic flux distribution of sine-wave shape can be expressed as follows.



$$\begin{pmatrix} v_u \\ v_v \\ v_w \end{pmatrix} = R_a \begin{pmatrix} i_u \\ i_v \\ i_w \end{pmatrix} + p \begin{pmatrix} \phi_u \\ \phi_v \\ \phi_w \end{pmatrix}$$

$$\begin{pmatrix} \phi_u \\ \phi_v \\ \phi_w \end{pmatrix} = \begin{pmatrix} L_u & M_{uv} & M_{wu} \\ M_{uv} & L_v & M_{vw} \\ M_{wu} & M_{vw} & L_w \end{pmatrix} \begin{pmatrix} i_u \\ i_v \\ i_w \end{pmatrix} + \psi \begin{pmatrix} \cos\theta \\ \cos(\theta - 2\pi/3) \\ \cos(\theta + 2\pi/3) \end{pmatrix}$$

v_u, v_v, v_w : Each phase armature voltage

L_u, L_v, L_w : Each phase self inductance

i_u, i_v, i_w : Each phase armature current

M_{uv}, M_{vw}, M_{wu} : Each phase mutual inductance

ϕ_u, ϕ_v, ϕ_w : Each phase armature interlinkage flux

ψ : Maximum value of armature interlinkage flux depending on permanent magnet

R_a : Each phase armature resistance

θ : Lead angle of permanent magnet (rotor) from U phase

p : Differential operator

Figure 3-1 Conceptual diagram of the three phase permanent magnetic synchronous motor

Here, self-inductance and mutual inductance are expressed as shown in the following formula.

$$\begin{cases} L_u = l_a + L_a - L_{as}\cos(2\theta) \\ L_v = l_a + L_a - L_{as}\cos(2\theta + 2\pi/3) \\ L_w = l_a + L_a - L_{as}\cos(2\theta - 2\pi/3) \end{cases}$$

$$\begin{cases} M_{uv} = -L_a/2 - L_{as}\cos(2\theta - 2\pi/3) \\ M_{vw} = -L_a/2 - L_{as}\cos 2\theta \\ M_{wu} = -L_a/2 - L_{as}\cos(\theta + 2\pi/3) \end{cases}$$

l_a : Leakage inductance for one phase

L_a : Average value of effective inductance for one phase

L_{as} : Amplitude of effective inductance for one phase

3.2 Vector control

The d axis is set in the direction of the magnetic flux (N pole) of the permanent magnet and the q axis is set in the direction which progresses by 90 degrees from the d axis. Then by using the following conversion matrix, coordinate conversion is performed.

$$C = \sqrt{\frac{2}{3}} \begin{pmatrix} \cos\theta & \cos(\theta - 2\pi/3) & \cos(\theta + 2\pi/3) \\ -\sin\theta & -\sin(\theta - 2\pi/3) & -\sin(\theta + 2\pi/3) \end{pmatrix}$$

$$\begin{pmatrix} v_d \\ v_q \end{pmatrix} = C \begin{pmatrix} v_u \\ v_v \\ v_w \end{pmatrix}$$

The voltage equation in the dq coordinate system is obtained as follow.

$$\begin{pmatrix} v_d \\ v_q \end{pmatrix} = \begin{pmatrix} R_a + pL_d & -\omega L_q \\ \omega L_d & R_a + pL_q \end{pmatrix} \begin{pmatrix} i_d \\ i_q \end{pmatrix} + \begin{pmatrix} 0 \\ \omega \psi_a \end{pmatrix}$$

v_d, v_q : Each phase armature voltage

L_d, L_q : Each phase self inductance

i_d, i_q : Each phase armature current

$L_d = l_a + 3/2(L_a - L_{as})$、 $L_q = l_a + 3/2(L_a + L_{as})$

ψ_a : Value of armature interlinka ge flux depending on permanent magne

R_a : Each phase armature resistance

$\psi_a = \sqrt{3/2}\psi$

Based on this, it can be assumed that 3 phase alternating current is 2 phase direct current.

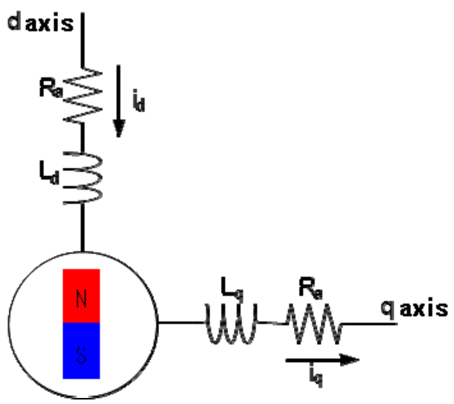


Figure 3-2 Conceptual diagram of the two phase direct current motor

Size of the torque generated in the motor can be obtained as follows from the exterior product of the electric current vector and armature interlinkage magnetic flux. The first term on the right side of this formula is called magnetic torque and the second term on the right side of this formula is the reluctance torque.

$$T = P_n \{\psi_a i_q + (L_d - L_q) i_d i_q\}$$

T : Motor torque P_n : Number of pole pairs

The motor which has no difference between the d axis and q axis inductance is defined as a motor which does not have saliency. In this case, as the reluctance torque is 0, the torque increases proportionally to the q axis current. Due to this, the q axis current is called torque current. On the other hand, d axis current is sometimes called excitation current, because it can be assumed that the d axis current increase is the magnetic flux of permanent magnet decrease for q axis voltage.

As SPMSM generally does not have saliency, the d axis current unnecessary for generating torque is controlled to while controlling the speed. This is known as id = 0 control. On one hand, the motion equation of the motor in this case is expressed as follows. This equation shows that motor speed is increased by increasing the q axis current.

$$I \frac{d\omega}{dt} = P_n \psi_a i_q - T_L$$

T_L : Load torque I : Numberof polepairs

This system uses not motion equation but PI control for speed control. The q axis current command value is calculated by the following formula.

$$i_q^* = (K_{P\omega} + \frac{K_{I\omega}}{s})(\omega^* - \omega)$$

$K_{P\omega}$: Speed PI ratio gain $K_{I\omega}$: Speed P I integral gain s : Laplace operator

To achieve early stabilization, the PI control is also used for the d axis and q axis current values. A command voltage value is acquired by current PI control.

$$v_d^* = (K_{Pi_d} + \frac{K_{Ii_d}}{s})(i_d^* - i_d)$$

K_{Pi_d} : d axis current PI propotional gain K_{Ii_d} : d axis current PI integral gain

$$v_q^* = (K_{Pi_q} + \frac{K_{Ii_q}}{s})(i_q^* - i_q)$$

K_{Pi_q} : q axis current PI propotional gain K_{Ii_q} : q axis current PI integral gain

Inductive voltage is generated when the motor is rotated. The effect on d axis voltage due to q axis current and on q axis voltage due to d axis current and magnetic flux of permanent magnet becomes significant along with the increase in speed. This d axis and q axis interference may delay the stability of a current value. In order to avoid this, the voltage of each axis is calculated by performing feed forward so that the interference term of each axis can be canceled beforehand.

$$v_d^* = (K_{Pi_d} + \frac{K_{Ii_d}}{s})(i_d^* - i_d) - \omega L_q i_q$$

$$v_q^* = (K_{Pi_q} + \frac{K_{Ii_q}}{s})(i_q^* - i_q) + \omega(L_d i_d + \psi_a)$$

This method to eliminate the effect of the interference term is known as decoupling control. This enables to control the d axis and q axis independently.
Vector control is a method by which the 3 phase alternating current motor is converted to the 2 phase direct current motor that can be controlled each phase (d,q) independently while managing the position, speed and torque of the rotor.

Control flow of the vector control is shown below.

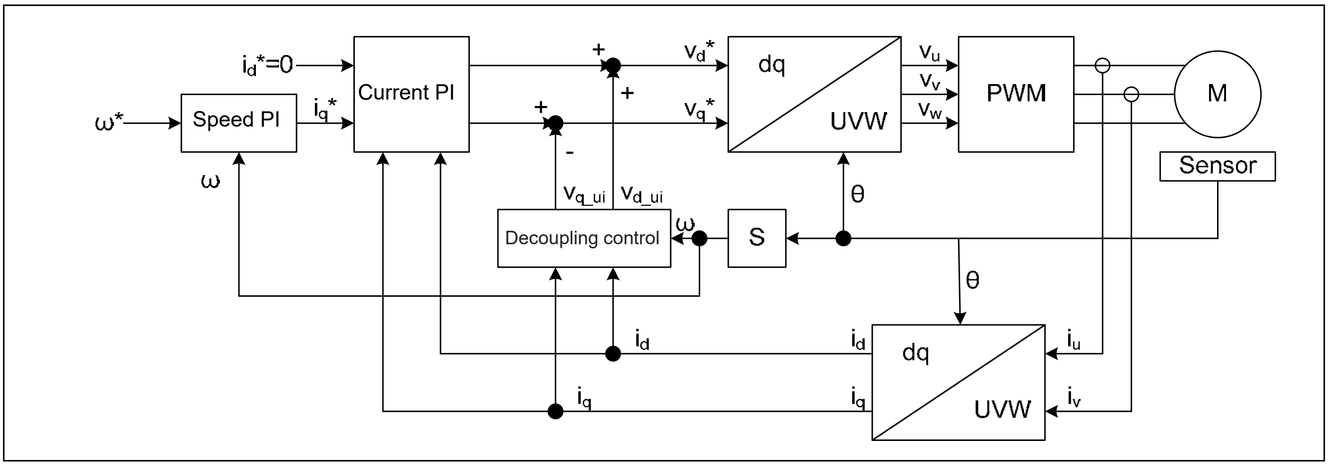


Figure 3-3 Control Flow of the Vector Control

3.3 Sensorless vector control based on the current estimation error

For the vector control, position sensors of the encoder and resolver etc are required as voltage is set according to the rotor position. When the position sensors are not used, in other words, in the case of the sensorless vector control, it is necessary to estimate the position by some methods. These days, the demand for motor control by sensorless has increased and several methods are provided for estimating the position This part introduces the sensorless vector control used in this system, which is using current estimation error.

Position of the d axis is not clear as the position information of the actual motor is not available. As shown in the below figure, when γ axis is set in the location which lags behind by Δθ from the d axis and δ axis is set in the location 90 degrees ahead of the γ axis, the conversion formula from d q axis to the γ δ axis can be indicated as follows.

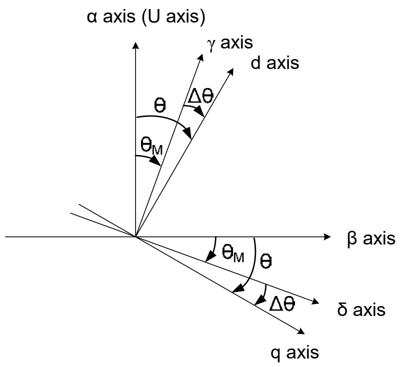


Figure 3-4 Relation between d q axis and γ δ axis

$$\begin{pmatrix} \gamma \\ \delta \end{pmatrix} = \begin{pmatrix} \cos\Delta\theta & \sin\Delta\theta \\ -\sin\Delta\theta & \cos\Delta\theta \end{pmatrix} \begin{pmatrix} d \\ q \end{pmatrix}$$

The equation in which above is applied to the SPMSM voltage equation and written in the electric current state equation format is as follows.

$$p\begin{pmatrix} i_\gamma \\ i_\delta \end{pmatrix} = -\begin{pmatrix} \frac{R}{L} & -\dot{\theta}_M \\ \dot{\theta}_M & \frac{R}{L} \end{pmatrix} \begin{pmatrix} i_\gamma \\ i_\delta \end{pmatrix} + \frac{1}{L}\begin{pmatrix} v_\gamma \\ v_\delta \end{pmatrix} - \frac{K_E\dot{\theta}}{L}\begin{pmatrix} -\sin\Delta\theta \\ \cos\Delta\theta \end{pmatrix}$$

Discretization is performed by using backward differential approximation (Euler's approximation) to this state equation.

$$\begin{pmatrix} i_\gamma(n) \\ i_\delta(n) \end{pmatrix} = \begin{pmatrix} i_\gamma(n-1) \\ i_\delta(n-1) \end{pmatrix} + \frac{T}{L}\left\{ \begin{pmatrix} v_\gamma(n-1) \\ v_\delta(n-1) \end{pmatrix} - R\begin{pmatrix} i_\gamma(n-1) \\ i_\delta(n-1) \end{pmatrix} - \dot{\theta}_M(n-1)L\begin{pmatrix} -i_\delta(n-1) \\ i_\gamma(n-1) \end{pmatrix} - e(n-1)\begin{pmatrix} -\sin\Delta\theta(n-1) \\ \cos\Delta\theta(n-1) \end{pmatrix} \right\}$$

$$\because e(n-1) = K_E\dot{\theta}(n-1)$$

As a motor model here, given that the motor parameters are written as R_M, L_M and e_M which are sufficiently equal to motor parameters of an actual motor and Δθ is set to 0, the current value at a sample point n can be represented as follows.

$$\begin{pmatrix} i_{\gamma M}(n) \\ i_{\delta M}(n) \end{pmatrix} = \begin{pmatrix} i_\gamma(n-1) \\ i_\delta(n-1) \end{pmatrix} + \frac{T}{L_M}\left\{ \begin{pmatrix} v_\gamma(n-1) \\ v_\delta(n-1) \end{pmatrix} - R_M \begin{pmatrix} i_\gamma(n-1) \\ i_\delta(n-1) \end{pmatrix} - \dot{\theta}_M(n-1) L_M \begin{pmatrix} -i_\delta(n-1) \\ i_\gamma(n-1) \end{pmatrix} - e_M(n-1)\begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}$$

Depending on the difference between actual motor current and motor model current, the current estimation error can be indicated as follows.

$$\begin{pmatrix} \Delta i_\gamma(n) \\ \Delta i_\delta(n) \end{pmatrix} = \frac{T}{L}\begin{pmatrix} e(n-1)\sin\Delta\theta(n-1) \\ e_M(n-1) - e(n-1)\cos\Delta\theta(n-1) \end{pmatrix}$$

When Δθ is sufficiently small, the current estimation error can be approximated as follows.

$$\begin{pmatrix} \Delta i_\gamma(n) \\ \Delta i_\delta(n) \end{pmatrix} \approx \frac{T}{L}\begin{pmatrix} e(n-1)\Delta\theta(n-1) \\ -\Delta e(n-1) \end{pmatrix}$$

$$\Delta e(n-1) = e(n-1) - e_M(n-1)$$

If both Δe and Δθ are 0, it can be considered that the actual model is synchronized with the motor model. e_M is estimated by feeding back Δi_δ such that Δe becomes 0. Similarly, the θ_M value is estimated by feeding back Δi_γ such that Δθ becomes 0. The motor model is thus matched with the actual model.
The e_M estimation equation can be expressed as follows.

$$e_M(n) = e_M(n-1) - K_e \Delta i_\delta(n)$$

Here, K_e is the speed electromotive force gain. Similarly, the θM estimation equation can be written as follows.

$$\theta_M(n) = \theta_M(n-1) + \frac{T}{K_{EM}} e_M(n) + K_\theta \operatorname{sgn}\{\dot{\theta}_M(n-1)\}\Delta i_\gamma(n)$$

$$\operatorname{sgn}\{\dot{\theta}_M(n-1)\} = \begin{cases} 1 & ;\ \dot{\theta}_M(n-1) \ge 0 \\ -1 & ;\ \dot{\theta}_M(n-1) < 0 \end{cases}$$

Here, K_{EM} is the electromotive force coefficient of the motor model and K_θ is the position estimation gain. Also, $p\theta_M$ sign is used instead of the pθ sign.

$$\dot{\theta}_M = \frac{1}{T}\{\theta_M(n) - \theta_M(n-1)\} = \frac{e_M}{K_{EM}} + \Delta\dot{\theta}_M(n)$$

$$\Delta\dot{\theta}_M(n) = \frac{K_\theta}{T}\operatorname{sgn}\{\dot{\theta}_M(n-1)\}\Delta i_\gamma(n)$$

In the control, LPF for the speed correction term is used as follows. Here, 0 < K < 1.

$$\dot{\theta}_{Mo}(n)=\frac{e_M(n)}{K_{EM}}+\Delta\dot{\theta}_{Mo}(n)$$
$$\Delta\dot{\theta}_{Mo}(n)=\Delta\dot{\theta}_{Mo}(n-1)+K\{\Delta\dot{\theta}_M(n)-\Delta\dot{\theta}_{Mo}(n-1)\}$$

Control flow of this control method is shown below.

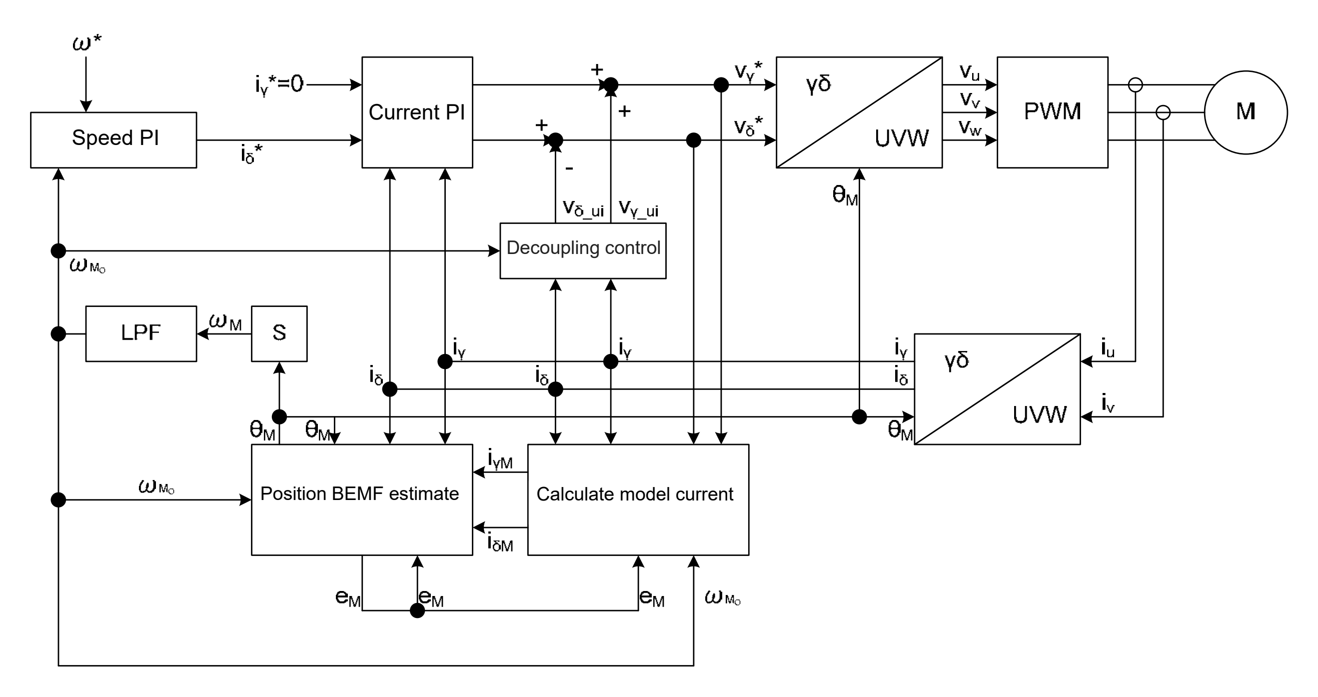


Figure 3-5 Control flow of the sensorless vector control based on the current estimation error method

3.4 Triangular wave comparison method

In order to actually output the voltage command value, the triangular wave comparison method which determines the pulse width of the output voltage by comparing the carrier waveform (triangular wave) and voltage command value waveform is used. By using this PWM formula, output of the voltage command value of the pseudo sinusoidal wave can be performed.

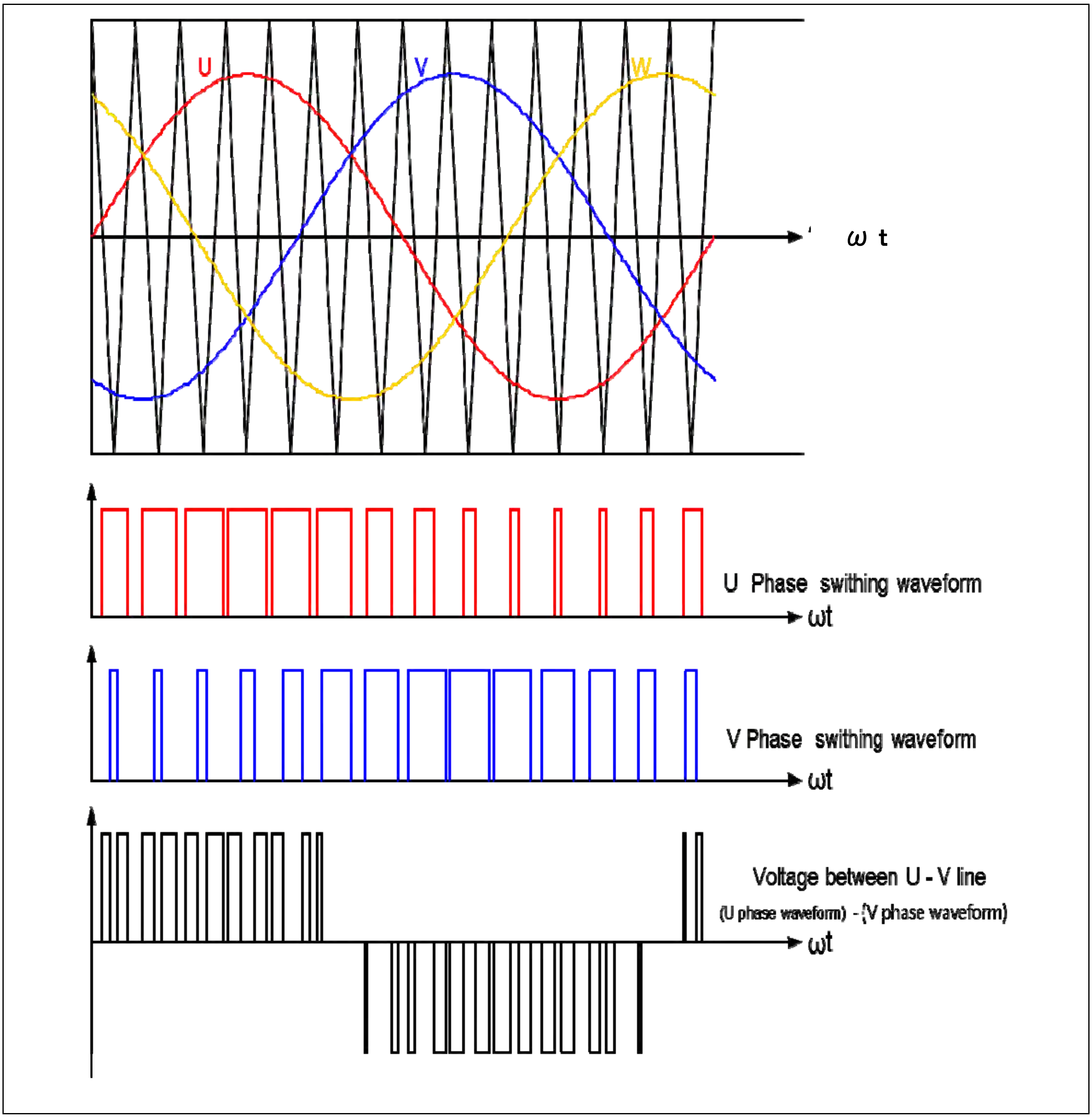


Figure 3-6 Conceptual diagram of the triangular wave comparison method

Here, as shown in the Figure 3-7, ratio of the output voltage pulse to the carrier wave is called as duty.

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

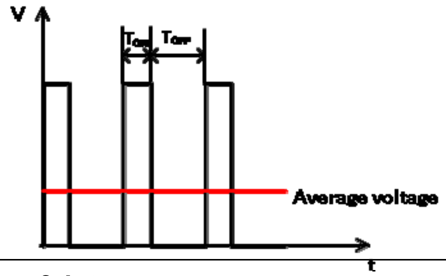


Figure 3-7 Definition of duty

Modulation factor m is defined as follows.

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

m: Modulation factor
V: Reference Voltage
E: Inverter Voltage

A request control can be performed by setting this modulation factor on the register which determines PWM duty.

4. Description of the control program

Control program of this system is explained here.

4.1 Contents of Control

4.1.1 Motor start/stop

Starting and stopping of the motor are controlled by input from SW1.

A general-purpose port (P31) is assigned to SW1. The P31 port is read within the TAU(1ms interrupt). When P31 is at a “Low” level, it is determined that the start switch is being pressed. Conversely, when the level is switched to “High”, the program determines that the motor should be stopped.

(1) Inverter bus voltage

Inverter bus voltage is measured as given in Table 4-1.
It is used for modulation factor calculation and over voltage detection
(When an abnormality is detected, PWM is stopped).

Table 4-1 Inverter Bus Voltage Conversion Ratio

Item	Conversion ratio (Inverter bus voltage V_{dc}: A/D conversion value)	channel
Inverter bus voltage	0 [V]～25 [V]：0000H～03FFH	ANI8

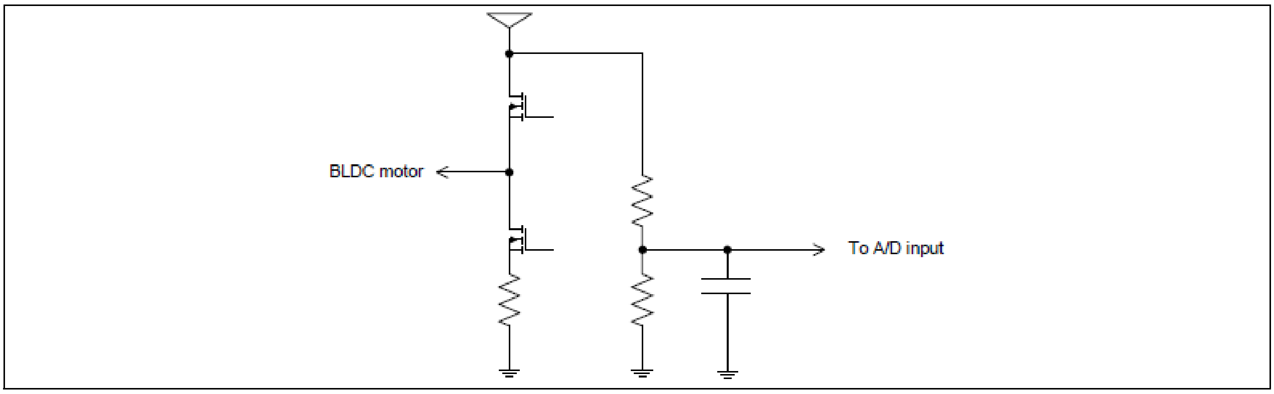


Figure 5-3 Conceptual diagram of inverter voltage measurement external circuit

(2) U, V, W phase current

The U, V, W phase currents are measured as shown in Table 4-2 and used in vector control.

Table 4-2 Conversion Ratio of U and W Phase Current

Item	Conversion ratio (U, V, W phase current: A/D conversion value)	チャネル
U,V,W phase current	-83.3 [A]～83.3 [A]：0000H～03FFH	ANI2,ANI3,ANI4

4.1.2 Control method

The motor is driven in an open loop at the time of startup. After a fixed time has passed, the motor is driven by the sensorless vector control based on the current estimation error explained in chapter 3 (please refer to the block diagram in Figure 3-5). PI control is used to control the speed.

4.1.3 System protection function

This control program has the following three types of error status and executes emergency stop functions in case of occurrence of respective errors.

• Over current error

High impedance output is made to the PWM output port in response to an emergency stop signal (over current detection) from the hardware (emergency stop without involving CPU). The INTP0 port is used.
In addition, U, V, and W phase currents are monitored by 187.5 [μs] intervals. When an over current (when the current exceeds 15 [A]) is detected, the CPU executes emergency stop.

・Over voltage error

The inverter bus voltage is monitored by 187.5 [us] intervals. When an over voltage is detected (when the voltage exceeds 24 [V]), the CPU performs emergency stop. Here, the over voltage limit value 24 [V] is set by considering the error of resistance value and error of supply voltage by AC adapter etc.

• Low voltage error

The inverter bus voltage is monitored by 187.5 [us] intervals. The CPU performs emergency stop when low voltage (when voltage falls below 3 [V]) is detected.

4.2 Function Specifications

Multiple control functions are used in this control program. Lists of control functions are given below. For detailed processing, please refer to flowcharts or source files.

Table 4-4 Control functions List(1/5)

File name	Function name	Process overview
foc_function.c	focInitState Input : None Output : None	Initialize status and variable
	focExternalErrorDetected Input : (uint8)errType Output : None	Detect error handling of HW
	focCalcFieldOrientedControl Input : None Output : None	Current control loop with FOC
	focGetStatus Input : None Output : (uint8)focStatus	Detect status of current control with FOC
	focGetErrorType Input : None Output : (uint8)focErrorFlag	Detect error of current control with FOC
	focPiCtrl Input : (ST_PI_CTRL) *obbj Output : (int16)result	PI control function.
	focLimit16 Input : (int16)srcValue (uint16)limValue Output : (int16)retValue	Limit function
	focCompensateAngle Input : (int16)angleRad (int16)speedRad (uint16)compTime Output : (int16)retVal	Compensation degree position
	focCalcUw2dq Input : (int16)srcU (int16)srcW (int16)refRad Output : (int16)*dstD (int16)*dstQ	dq fixed coordinate transformation function
	focCalcDq2ab Input : (int16)srcD (int16)srcQ (int16)refRad Output : (int16)*dstA (int16)*dstB	dq rotation coordinate transformation function
	focCalcAb2Uvw Input : (int16)srcA (int16)srcB Output : (int16)*dstU (int16)*dstV (int16)*dstW	Two axis to three axis coordinate transformation
	focThreeShuntSamplingCall Input : (int16)refVu (int16)refVv (int16)refVw Output : (int16)retVal	Determine Sampling phase function

Table 4-4 Control function List（2/5）

File name	Function name	Process overview
main.c	main Input: None Output: None	Initialize variable in main operation
rl78_interrupt.c	_INTP0_Interrupt Input: None Output: None	External interrupt
	_TRD0_Interrupt Input: None Output: None	Timer RD0 interrupt
	_TRD1_Interrupt Input: None Output: None	Timer RD1 interrupt
	_TAU0_Interrupt Input: None Output: None	Timer TAU0 interrupt

Table 4-4 Control function List（3/5）

File name	Function name	Process overview
rl78f14_inv.c	invInitBoad Input：(uint16)invCtrlTimer (uint16)invPwmCarrier Output：None	Initialize board
	invPdInit Input：None Output：None	Initialize setting of R2A25108 Set TRD
	invPdStart Input：None Output：None	Start output of Pre-driver
	invPdStop Input：None Output：None	Stop output of Pre-driver
	invSetUVW Input：(int16)refVu (int16)refVv (int16)refVw Output：None	Set Duty ratio to TRD from reference voltage
	invGetError Input：None Output：None	Check error from R2A25108
	invAdInit Input：None Output：None	Initialize A/D converter
	invAdGetIu,invAdGetIv,invAdGetIw Input：None Output：None	Detect voltage of inverter
	invAdGetVu,invAdGetVv,invAdGetVw Input：None Output：None	Detect current of inverter
	invAdGetVpn Input：None Output：None	Detect voltage of terminal
	invAdGetIdc Input：None Output：None	Detect current of terminal
	invAdGetTemp Input：None Output：None	Detect voltage of Temp terminal
	int16_t invAdGetIdc3 Input：None Output：None	Detect consumption current of motor
	invMuteOn Input：None Output：None	Set ON Mute function for R2A25108
	invMuteOff Input：None Output：None	Set Off Mute function for R2A25108
	invSWGetDipSW Input：None Output：INV_PORT_ALLDIPSW()	Detect status of dip-switch
	invSWGetPushSw Input：None Output：INV_PORT_PSHSW()	Detect status of push-switch

Table 4-4 Control function List (4/5)

File name	Function name	Process overview
rl78f14_mcu.c	mcuCpuInit Input : None Output : None	Initialize setting of clock
	mcuIoInit Input : None Output : None	Initialize I/O ports
	mcuTauInit Input : (uint16)time_us Output : None	Initialize TAU0 for interval timer
	mcuTrdInit Input : (uint8)trdPhase (uint8)trdInt (uint16)trdCarrierFreq (uint16)trdDeadtimeNs Output : None	Initialize TRD
	mcuAdcInit Input : None Output : None	Initialize A/D converter
	mcuTauStart Input : None Output : None	Start TAU0
	mcuTauStop Input : None Output : None	Stop TAU0
	mcuTrdStartCount Input : None Output : None	Start TRD counter
	mcuTrdStopCount Input : None Output : None	Stop TRD counter
	mcuTrdEnableOutput Input : None Output : None	Enable output of TRD
	mcuTrdDisableOutput Input : None Output : None	Disable output of TRD
	mcuTrdSetDuty Input : (int16)pwm1 (int16)pwm2 (int16)pwm3 Output : None	Change compare value for TRD
	mcuTrdClearIf Input : None Output : None	Clear flag for TRD interrupt
	mcuTrdClearWdt Input : None Output : None	Clear WDT counter
	mcuAdcGetData Input : (uint8)adcChanel Output : None	Acquisition of channel data for A/D converter
	mcuExternalInterrupt Input : None Output : None	Initialize External interrupt
	mcuIntp0ClearIf Input : None Output : None	Clear flag for INTP0

Table 4-4 Control function List（5/5）

File name	Function name	Process overview
Sequence.c	seqInitSetting Input：None Output：None	Initialize status
	seqExecEvent Input:(uint8)reqEvent Output:None	Run event
	seqActRun Input:(uint8)curState Output:ret	Run starting event with motor
	seqActStop Input:(uint8)curState Output:ret	Run stopping event with motor
	seqActNone Input:(uint8)curState Output:ret	No event with motor
	seqActReset Input:(uint8)curState Output:ret	Reset event with motor
	seqActError Input:(uint8)curState Output:ret	Set error event with motor
	seqGetSeqMode Input:None Output:seqModeSystem	Detect sequence mode
	seqGetErrorType Input：None Output：seqErrorStatus	Detect error status
user_control.c	userInitControlParam Input：None Output：None	Set parameter of each control
	userCurrentReferenceControl Input：None Output：None	Control reference current
	userControlTimer Input：None Output：None	Call interrupt of speed control
	userSequenceControl Input:(uint8)*ControlReqest Output:None	Change control parameter
	userSpeedControl Input：None Output：None	Control speed loop

4.3 Variables list

Lists of variables used in this control program are given below. Note that local variables are not described

Table 4-5 variables value List(1/2)

category	Variable name	type	content	Remark
Adjustment A/D converter offset	focTimeSettingOffset	uint16	Three phase current offset count value	
	focCurrentOffsetLpfK	int16	Three phase current offset filter factor	
	focTimeCountOffset	uint16	Current offset counter	
Current-related	focIdRef	int16	d axis current reference	
	focIqRef	int16	q axis current reference	
	focCurrentIu	int16	U phase current value	
	focCurrentIv	int16	V phase current value	
	focCurrentIw	int16	W phase current value	
	focCurrentId	int16	d axis current value	
	focCurrentIq	int16	q axis current value	
	focOffsetIu	int16	U phase current offset value	
	focOffsetIv	int16	V phase current offset value	
	focOffsetIw	int16	W phase current offset value	
	focCurrentDQLpfK	int16	dq axis current low pass filter value	
Voltage-related	focVdRef	int16	Vd voltage reference value	
	focVqref	int16	Vq voltage reference value	
	focMuRef	int16	U phase current reference value	
	focMvRef	int16	V phase current reference value	
	focMwRef	int16	W phase current reference value	
	focVdqLimit	int16	dq axis voltage Limit value	
Rotor angle position	focAngleRad	int16	Rotor angle position	Electrical[rad]
	focSpeedRad	int16	Motor speed	Electrical [rad/s]
Angle compensation	focCompTimeAd	int16	A/D converter sampling time	
	focCompTimePWM	int16	Motor control PWM time	
Status	focStatus	uint8	Motor status	
	focErrorFlag	uint8	Error Flag	
Calculate buffer	focCalcBuf	uint8	Calculation buffer	

Table 4-5 Variable value List(2/2)

category	Variable name	type	content	Remark
Calculate buffer	invLastVdcVal	int16	Power supply voltage buffer	
	invMaxDuty	int16	Duty max calculation buffer	
TRD Timer	trdPeak	uint16	Timer RD Sin wave limit value	
Sequence status	seqModeSystem	uint8	Sequence status	
	seqErrorStatus	uint8	Sequence Error status	
Motor speed control	userRunMode	uint16	Speed control status	
	userRpmRef	int16	Speed reference value	[rpm]
	userRpmEst	int16	Speed calculation value	[rpm]
	userRpmRefRequest	int16	Input motor speed for user	[rpm]
	userRpmOltoFoc	int16	Speed value from open-loop to close-loop	
	userRpmSlope	int16	Accel slope	[rpm/ms]
	userSpeedRadRef	int16	Speed reference for user	[rad/s]
Current reference	userDelaySettingOlToFoc	int16	Wait close-loop control time	
	userIdRefOlRequest	int16	d axis reference current for open-loop	
	userIqRefOlRequest	int16	q axis reference current for open-loop	
	userIdRefVecRequest	int16	Reference voltage for FOC	
	userIdRefEnhRequest	int16	Reference voltage for FOC	
	userIdSlopeOl	int16	d axis slope current for open-loop	
	userIqSlopeOl	int16	q axis slope current for open-loop	
	userIdSlopeUpVec	int16	Id current up slope	
	userIdSlopeDownVec	int16	Id current down slope	

4.4 Macro definitions

Lists of macro definitions used in this control program are given below.

Table 4-6 Macro definitions list (1/3)

File name	Macro name	Definition value	Remark
control_parameter.h	CP_FREQ_SPEED	1000	Speed control frequency[Hz]
	CP_RPM_MAX_SPEED	6000	MAX speed value[rpm]
	CP_RPM_MIN_SPEED	800	MIN speed value[rpm]
	CP_RPM_SLOPE_REQ	6000	Accel speed value[rpm/s]
	CP_RPM_OL_TO_FOC	600	Change speed value from open-loop to close-loop[rpm]
	CP_DELAY_OL_TO_FOC	100	Wait time for close-loop control[ms]
	CP_RPM_ENHANCE	1600	Allow speed value for d-axis control [rpm]
	CP_ID_REF_OL_REQ	2.2f	d-axis reference current for open-loop control [A]
	CP_IQ_REF_OL_REQ	0.0f	q-axis reference current for open-loop control[A]
	CP_ID_SLOPE_OL_REQ	8.0f	d axis slope current for open-loop[A/S]
	CP_IQ_SLOPE_OL_REQ	1.0f	q axis slope current for open-loop[A/S]
	CP_ID_REF_FOC_REQ	0.0f	d axis slope current for close-loop[A]
	CP_ID_REF_ENH_REQ	1.5f	q axis slope current for close-loop[A]
	CP_ID_SLOPE_UP_REQ	5.0f	MAX value of d axis slope current for close-loop(Up) [A/S]
	CP_ID_SLOPE_DOWN_REQ	4.5f	MAX value of d axis slope current for close-loop(Down) [A/S]
	CP_SPEED_PI_KP	1.5f	Constant of proportion for speed control
	CP_SPEED_PI_KI	0.002f	Constant of integration for speed control
	CP_IQ_LIMIT	7.0f	MAX reference current of q axis [A]
	CP_PWM_CARRIER	16000	Carrier frequency [Hz]
	CP_TIME_OFFSET	1.5	Adjust time for offset of three phase current [s]
	CP_DECIMATION	2	Decimation control cycle time
	CP_ID_PI_KP	0.18385f	Constant of proportion for d-axis current control
	CP_ID_PI_KI	0.01529f	Constant of integration for d-axis current control
	CP_IQ_PI_KP	0.18640f	Constant of proportion for q-axis current control
	CP_IQ_PI_KI	0.01597f	Constant of integration for q-axis current control
	CP_CURRENT_OFFSET_FACTO R	0.025f	Adjust filter factor of three phase current
	CP_CURRENT_DQLPF_FACTO R	0.25f	dq axis Low pass filter factor
	CP_THETA_EST_K	0.050640f	Speed estimate factor
	CP_ANGLE_LPF_K	0.053941f	Angle estimate factor
	CP_EMF_EST_K	0.080437f	BEMF estimate factor
	CP_OFFSET_COUNT	(CP_PWM_CARRIER / (1 + CP_DECIMATION) * CP_TIME_OFFSET)	Counter value for adjust time of three phase current offset
	SCALE_SPEED_RAD(pp)	SC_FLOAT2INT16(((3. 14159265f * 2 * CP_RPM_MAX_SPEE D * pp)/60), F2I_SCALE)	Scaling angle rate exponent
	SCALE_ANGLE_RAD	SC_FLOAT2INT16((3.1 4159265f * 2), F2I_SCALE)	Scaling angle exponent

Table 4-6 Macro definitions list (2/3)

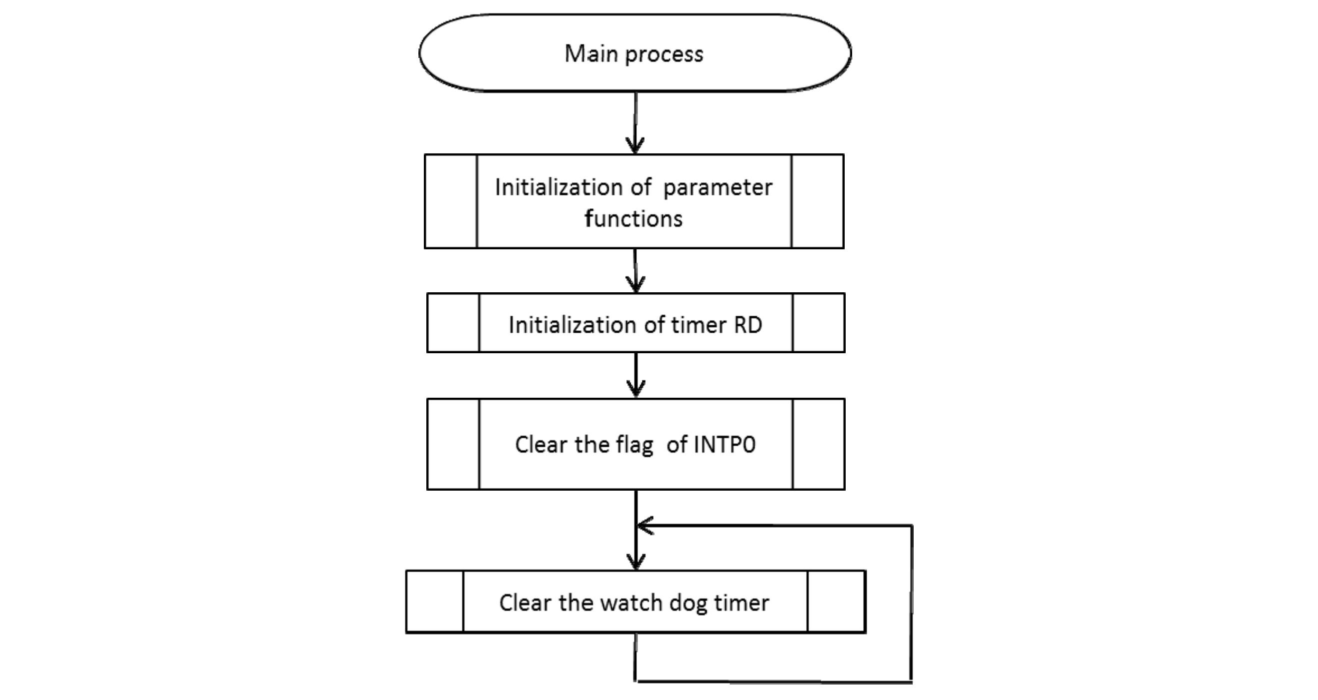
File name	Macro name	Definition value	Remark
foc_function.h	USE_FOCLIB	1	Use rl78_foclib functions or not
	FOC_STATE_ADJUST	0x00U	FOC control status
	FOC_STATE_READY	0x01U	
	FOC_STATE_ERROR	0xFFU	
	FOC_ERR_BASE	0xE0U	Error mode code 00:No error E0: Over Current error(HW Detection) E1:Over Current error(SW Detection) E2:Over voltage error E3:Low voltage error EF:Pre-Driver error
	FOC_ERR_NON	0x00U	
	FOC_ERR_OCD_HW	(FOC_ERR_BASE + 0x0U)	
	FOC_ERR_OCD	(FOC_ERR_BASE + 0x1U)	
	FOC_ERR_OVD	(FOC_ERR_BASE + 0x2U)	
	FOC_ERR_UVD	(FOC_ERR_BASE + 0x3U)	
	FOC_ERR_PDERR	(FOC_ERR_BASE + 0xFU)	
	MATH_PI	3.14159265f	pi constant
	MATH_TWOPI	(2.0f * MATH_PI)	2pi constant
	MATH_SQRT_3d2	1.224745f	√(3/2) constant
	MATH_SQRT_2d3	0.816497f	√(2/3) constant
	MATH_SQRT_3_2	0.866025f	√(3)/2 constant
	MATH_SQRT_2	1.414214f	√(2) constant
	MATH_SQRT_2_2	(MATH_SQRT_2 / 2)	√(2)/2 constant
	MATH_RPM2RADPS	(MATH_TWOPI/60.0f)	rpm-->rad/s Conversion factor from rotating speed to angle rate
	MATH_RADPS2RPM	(60.0f/MATH_TWOPI)	rad/s-->rpm Conversion factor from rotating speed to angle rate rotating speed
motor_parameter.h	MP_PP	4	Motor pole number
	MP_RA	0.075f	warring resister value [ohm]
	MP_LD	0.00009685f	Motor inductance Ld[H]
	MP_LQ	0.00010115f	Motor inductance Lq[H]
	MP_KE	0.00917f	BEMF motor constant[Vs/rad]
rl78_common.h	SFRBIT(sfr, bit)	(sfr ## . ## bit)	SFR bit access macro (for CA78K0R compiler)
		(sfr ## _bit.no ## bit)	SFR bit access macro (for CC-RL compiler)
	__EI()	EI()	Replace definition of built-in function (for CA78K0R compiler)
	__DI()	DI()	
	__halt()	HALT()	
	__stop()	STOP()	
	__brk()	BRK()	
	__nop()	NOP()	
rl78_interrupt.h	__interrupt		Replace definition of modifier (for CC-RL compiler)

Table 4-6 Macro definitions list (3/3)

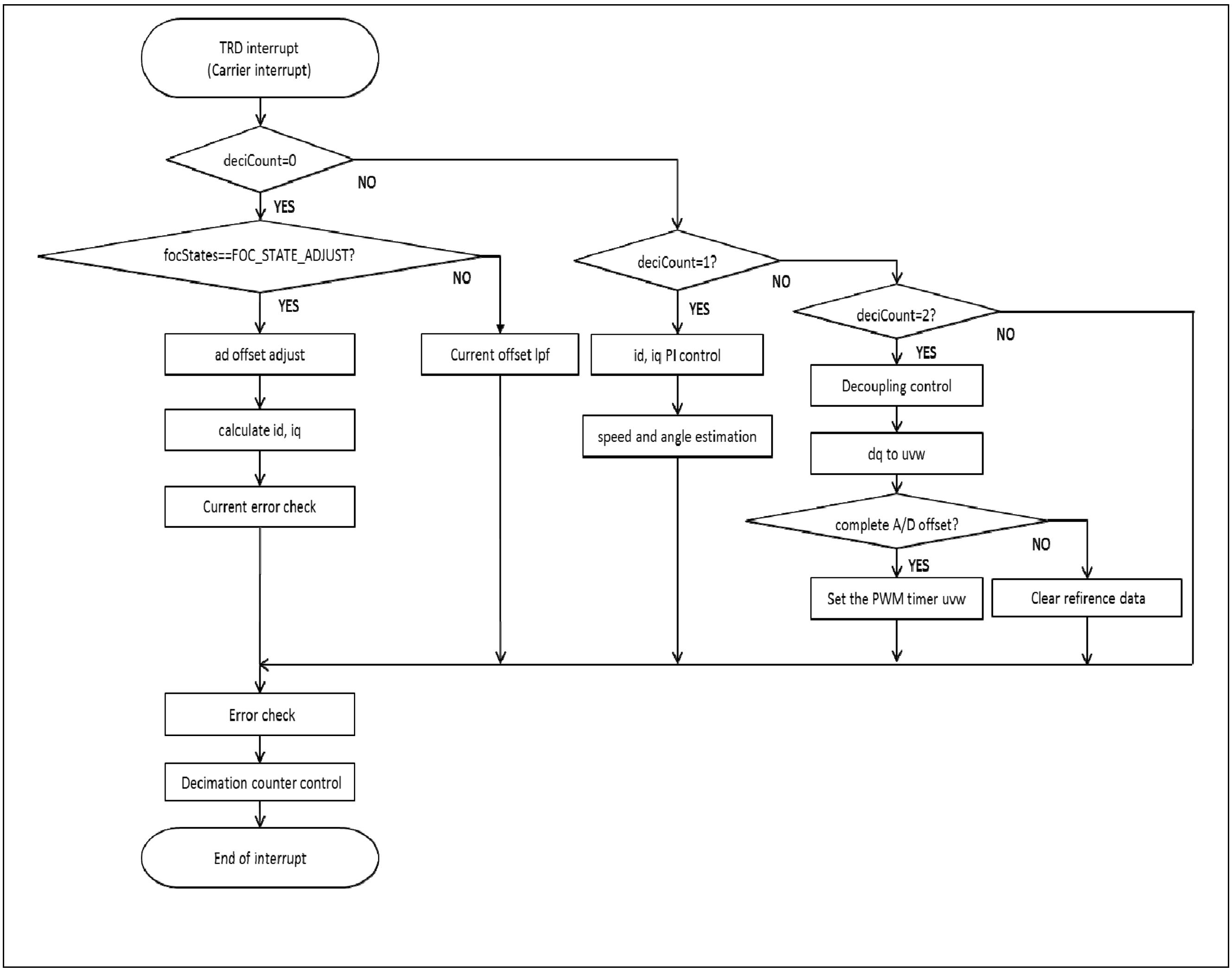
File name	Macro name	Definition value	Remark
rl78f14_inv.h	INV_ERR_BASE	0xC0	Error Mode code 00:No error C0:Over voltage error C1:Under voltage error CD: Power supply error CE: Desaturation error CF: Power supply low voltage error
	INV_ERR_NONE	0x00	
	INV_ERR_VDC_OVD	(INV_ERR_BASE + 0x0U)	
	INV_ERR_VDC_UVD	(INV_ERR_BASE + 0x1U)	
	INV_ERR_OVD_TSD	(INV_ERR_BASE + 0xDU)	
	INV_ERR_SCB_SCG	(INV_ERR_BASE + 0xEU)	
	INV_ERR_UVD	(INV_ERR_BASE + 0xFU)	
	INV_PWM_DEADTIME	2000	Inverter dead time[2us]
	INV_SHUNT_R	0.002f	Inverter shunt resister[2mΩ]
	INV_AMP_GAIN	15	AMP Gain [15k/1k]
	INV_AD2CUR	((2.5f/INV_AMP_GAIN)/ INV_SHUNT_R)/(0xFFC0/2))	Conversion factor from A/D converter data to phase current value
	INV_AD2IDC	(52.0f/0x03FF)	Conversion factor from A/D converter data to current value of power supply
	INV_AD2VPN	(25.0f/0x03FF)	Conversion factor from A/D converter data to voltage value of power supply
	INV_AD2VOL	(25.0f/0x03FF)	Conversion factor from A/D converter data to phase voltage value
	INV_RATED_VOLTAGE	12.0f	Rated voltage
	INV_OV_LEVEL	24.0f	Over voltage detect level[V]
	INV_UV_LEVEL	3.0f	Low voltage detect level[V]
	INV_VOLTAGE_MAX	40.0f	Scope limitation voltage[V]
	INV_OC_LEVEL	15.0f	Over current detect level[A]
	INV_CURRENT_MAX	40.0f	Scope limitation current[A]
rl78f14_mcu.h	TRD_PHASE_ACTIVE_H	0U	mcuTauInit() function setting Output High Active with Inverter
	TRD_PHASE_ACTIVE_L	(TRD_PHASE_ACTIVE_H + 1)	mcuTauInit() function setting Output Low Active with Inverter
	TRD_TRDINT_TOP	0U	mcuTauInit() function setting Set interrupt condition(TOP) of TRD timer
	TRD_TRDINT_BOTTOM	TRD_TRDINT_TOP + 1	mcuTauInit()function setting Set interrupt condition(Bottom) of TRD timer
	TRD_SET_PWM_MAX	0x4000	Scaling data MAX PWM duty rate
	TRD_SET_PWM_MAX_E	14	Scaling data PWM duty rate
	MCU_FIH_CLK	48	High speed oscillator frequency
	MCU_FCLK	24	MCU frequency
sequence.h	SEQ_MODE_STOP	0x00	System mode 00:Motor Stop mode 01:Motor Run mode 02:Motor Error mode
	SEQ_MODE_RUN	0x01	
	SEQ_MODE_ERROR	0x02	
	SEQ_SIZE_STATE	3	Status over threshold
	SEQ_EVENT_STOP	0x00	Event data 00:Stop 01:Run 02:Error 03:Reset
	SEQ_EVENT_RUN	0x01	
	SEQ_EVENT_ERROR	0x02	
	SEQ_EVENT_RESET	0x03	
	SEQ_SIZE_EVENT	4	Event status over threshold
	SEQ_ERR_NONE	0x00U	Error Status Initialize data
	SEQ_ERR_UNKNOWN	0xFFU	Occur Status error

4.5 Control flow (flow chart)

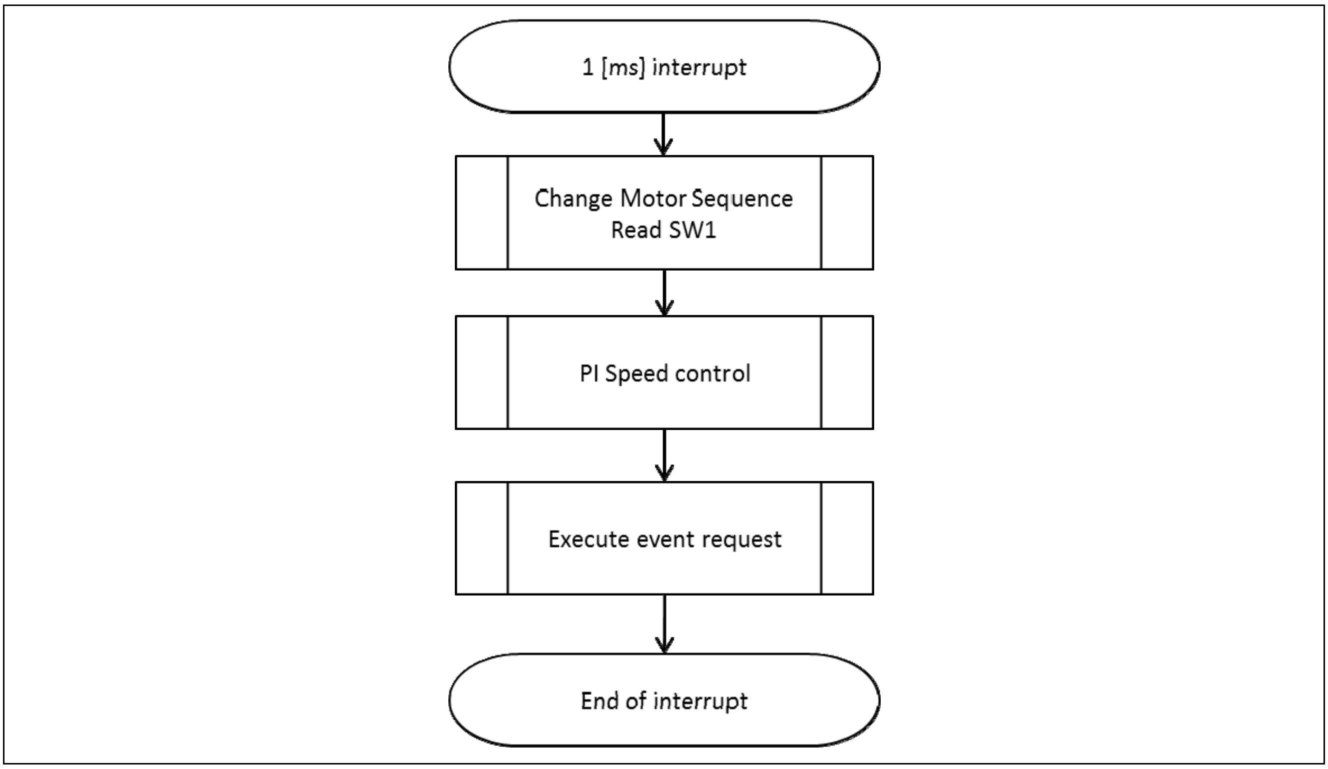
(1) Main process



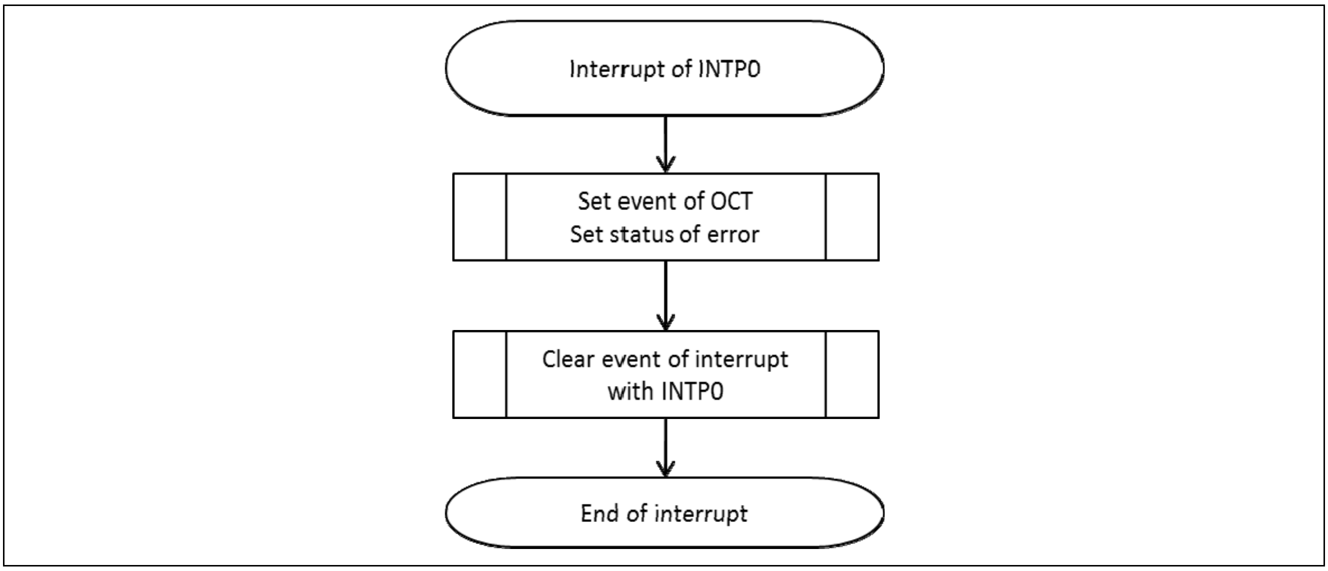
(2) Timer RD interrupt



(3) 1 [ms] TAU0 interrupt



(3) INTP0 interrupt



Website and Support

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

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

All trademark and registered trademarks are the property of their respective owners.

Rev.	Date	Description	
		Page	Summary
1.00	2017.3.31	—	First edition issued

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other disputes involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawing, chart, program, algorithm, application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics products.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.
   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (space and undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. When using the Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat radiation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions or failure or accident arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please ensure to implement safety measures to guard them against the possibility of bodily injury, injury or damage caused by fire, and social damage in the event of failure or malfunction of Renesas Electronics products, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures by your own responsibility as warranty for your products/system. Because the evaluation of microcomputer software alone is very difficult and not practical, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please investigate applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive carefully and sufficiently and use Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall not use Renesas Electronics products or technologies for (1) any purpose relating to the development, design, manufacture, use, stockpiling, etc., of weapons of mass destruction, such as nuclear weapons, chemical weapons, or biological weapons, or missiles (including unmanned aerial vehicles (UAVs)) for delivering such weapons, (2) any purpose relating to the development, design, manufacture, or use of conventional weapons, or (3) any other purpose of disturbing international peace and security, and you shall not sell, export, lease, transfer, or release Renesas Electronics products or technologies to any third party whether directly or indirectly with knowledge or reason to know that the third party or any other party will engage in the activities described above. When exporting, selling, transferring, etc., Renesas Electronics products or technologies, you shall comply with any applicable export control laws and regulations promulgated and administered by the governments of the countries asserting jurisdiction over the parties or transactions.
10. Please acknowledge and agree that you shall bear all the losses and damages which are incurred from the misuse or violation of the terms and conditions described in this document, including this notice, and hold Renesas Electronics harmless, if such misuse or violation results from your resale or making Renesas Electronics products available any third party.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.3.0-1 November 2016)

RENESAS

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL II Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

© 2017 Renesas Electronics Corporation. All rights reserved.

Colophon 6.0