Vector control for permanent magnetic synchronous motor with inductive sensor

For Renesas Flexible Motor Control Series

Abstract

This application note aims for an explanation of vector control for permanent magnetic synchronous motor with induction sensors by using functions of microcontroller, and how to use the motor control development support tool, ‘Renesas Motor Workbench’.

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

Operation checking device

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

- RA6T2 (R7FA6T2BD3CFP)
- RA6T3 (R7FA6T3BB3CFM)
- RA4T1 (R7FA4T1BB3CFM)

Target software

The following shows the target software for this application note:

- RA6T2_MCILV1_SPM_IS_FOC_E2S_V110
- RA6T3_MCILV1_SPM_IS_FOC_E2S_V100
- RA4T1_MCILV1_SPM_IS_FOC_E2S_V100
Contents

1. Overview ................................................................................................................................. 4

2. Development environment ....................................................................................................... 4
  2.1 Operation check environment ............................................................................................ 4
  2.2 Hardware specifications ......................................................................................................... 5
    2.2.1 Hardware configuration diagram .................................................................................. 5
    2.2.2 Hardware modification details ...................................................................................... 6
    2.2.3 User interface ................................................................................................................ 8
    2.2.4 Peripheral functions ..................................................................................................... 10
  2.3 Software configuration .......................................................................................................... 28
    2.3.1 Software file configuration .......................................................................................... 28
    2.3.2 Module configuration ..................................................................................................... 30

3. Descriptions of the control program ...................................................................................... 33
  3.1 Contents of control ............................................................................................................... 33
    3.1.1 Motor start/stop .......................................................................................................... 33
    3.1.2 A/D Converter ............................................................................................................ 33
    3.1.3 Modulation (current control module) ........................................................................... 34
    3.1.4 State transition ............................................................................................................ 35
    3.1.5 Start-up method ........................................................................................................... 36
    3.1.6 Generation of Position Profile ...................................................................................... 37
    3.1.7 Speed measurement ..................................................................................................... 38
    3.1.8 Error correction of inductive sensor .......................................................................... 39
    3.1.9 System protection function ......................................................................................... 41
    3.1.10 AD triggers ............................................................................................................... 42
  3.2 Function specifications of vector control with inductive sensors software ......................... 43
  3.3 Contents of control ............................................................................................................... 50
    3.3.1 Configuration Options ................................................................................................. 50
    3.3.2 Configuration Options for included modules ............................................................... 50
  3.4 Control flowcharts ............................................................................................................... 57
    3.4.1 Main process .............................................................................................................. 57
    3.4.2 Current Control Period Interrupt (Carrier synchronized Interrupt) Process ................. 58
    3.4.3 Speed Control Period Interrupt Process .................................................................... 59
    3.4.4 Over Current Detection Interrupt Process ................................................................. 60

4. Project Operation Overview ................................................................................................... 61
  4.1 Importing the Demo Project ............................................................................................... 61
  4.2 Building and Debugging ..................................................................................................... 62
  4.3 Quick Start ......................................................................................................................... 63
4.4 Motor Control Development Support Tool ‘Renesas Motor Workbench’ .............................................. 64
  4.4.1 Overview.............................................................................................................................................. 64
  4.4.2 Easy function operation example ........................................................................................................ 65
  4.4.3 List of variables for Analyzer function ................................................................................................. 69
  4.4.4 Operation Example for Analyzer ......................................................................................................... 70
  4.4.5 Tuner function...................................................................................................................................... 71
  4.4.6 Example of changing communication speed....................................................................................... 72
  4.4.7 How to use the built-in communication library ..................................................................................... 73

5. Reference Documents ........................................................................................................... 76

Revision History ............................................................................................................................ 77
Vector control for permanent magnetic synchronous motor with inductive sensor

1. Overview

This application note explains how to implement the vector control software that drives permanent magnetic synchronous motor (PMSM) with inductive sensors using the RA6T2 microcontroller and how to use the motor control development support tool, ‘Renesas Motor Workbench’.

Note that this software uses the vector control algorithm described in the application note ‘Vector control with encoder for permanent magnet synchronous motor (Algorithm)’ (R01AN3789), so please refer to that for the details of the algorithm.

2. Development environment

2.1 Operation check environment

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

Table 2-1 Hardware Development Environment

<table>
<thead>
<tr>
<th>Classification</th>
<th>Product used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microcomputer / CPU card P/N</td>
<td>RA6T2 (R7FA6T2BD3CFP) / RTK0EMA270C00000BJ</td>
</tr>
<tr>
<td></td>
<td>RA4T1 (R7FA4T1BB3CFM) / RTK0EMA430C00000BJ</td>
</tr>
<tr>
<td></td>
<td>RA6T3 (R7FA6T3BB3CFM) / RTK0EMA330C00000BJ</td>
</tr>
<tr>
<td>Inverter board</td>
<td>MCI-LV-1 / RTK0EM00000S04020BJ</td>
</tr>
<tr>
<td>motor</td>
<td>BLY171D-24V-4000 (manufactured by Anaheim Automation)</td>
</tr>
<tr>
<td>Sensor</td>
<td>Inductive sensor(Note3): IPS2200</td>
</tr>
</tbody>
</table>

Table 2-2 Software Development Environment

<table>
<thead>
<tr>
<th>e²studio version</th>
<th>FSP version</th>
<th>Toolchain version</th>
</tr>
</thead>
<tbody>
<tr>
<td>e²studio : 2023-04</td>
<td>V4.4.0</td>
<td>GCC ARM Embedded : V10.3.1.20210824</td>
</tr>
</tbody>
</table>

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

2.2.1 Hardware configuration diagram

![Hardware Configuration Diagram](image)

Figure 2-1 Hardware Configuration Diagram
2.2.2 Hardware modification details

Jumper pins need to be changed to use this system.

(1) RA6T2
Please change the connection of a jumper (JP4) to connect 2-3 pins from 1-2 pins.
Please change the connection of a jumper (JP17 and JP18) to connect 1-2 pins from 2-3 pins.
(2) RA4T1/RA6T3

Please change the connection of a jumper (JP1 and JP3) to connect 2-3 pins from 1-2 pins.
Please change the connection of a jumper (JP5 and JP6) to connect 1-2 pins from 2-3 pins.

Figure 2-3 Change the connection of a jumper
### 2.2.3 User interface

List of user interfaces of this system is given in Table 2-3.

**Table 2-3 User interface**

<table>
<thead>
<tr>
<th>Item</th>
<th>Interface component</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rotation speed command</td>
<td>Variable resistor (VR)</td>
<td>Reference value of rotation speed input (analog value)</td>
</tr>
<tr>
<td>START/STOP</td>
<td>Toggle switch (SW1)</td>
<td>Motor rotation start/stop command</td>
</tr>
<tr>
<td>ERROR RESET</td>
<td>Push switch (SW2)</td>
<td>Command of recovery from error status</td>
</tr>
<tr>
<td>LED1</td>
<td>Orange LED</td>
<td>- At the time of motor rotation: ON</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- At the time of stop: OFF</td>
</tr>
<tr>
<td>LED2</td>
<td>Orange LED</td>
<td>- At the time of error detection: ON</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- At the time of normal operation: OFF</td>
</tr>
<tr>
<td>LED3</td>
<td>Orange LED</td>
<td>- Positioning completed: ON</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Positioning not completed: OFF (Only RA6T2)</td>
</tr>
<tr>
<td>RESET</td>
<td>Push switch (RESET)</td>
<td>System reset</td>
</tr>
</tbody>
</table>

List of port interfaces of this system is given in Table 2-4.

**Table 2-4 Port Interfaces**

<table>
<thead>
<tr>
<th>Function</th>
<th>RA6T2</th>
<th>RA4T1</th>
<th>RA6T3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inverter bus voltage measurement</td>
<td>PA06 / AN006</td>
<td>P004 / AN004</td>
<td>P004 / AN004</td>
</tr>
<tr>
<td>For rotation speed command value</td>
<td>PB00 / AN008</td>
<td>P005 / AN005</td>
<td>P005 / AN005</td>
</tr>
<tr>
<td>(analog value)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>START/STOP toggle switch (SW1)</td>
<td>PD04</td>
<td>P304</td>
<td>P304</td>
</tr>
<tr>
<td>ERROR RESET push switch (SW2)</td>
<td>PD07</td>
<td>P200</td>
<td>P200</td>
</tr>
<tr>
<td>LED1 ON/OFF control</td>
<td>PD01</td>
<td>P113</td>
<td>P113</td>
</tr>
<tr>
<td>LED2 ON/OFF control</td>
<td>PD02</td>
<td>P106</td>
<td>P106</td>
</tr>
<tr>
<td>LED3 ON/OFF control</td>
<td>PD03</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U phase current measurement</td>
<td>PA04 / AN004</td>
<td>P000 / AN000</td>
<td>P000 / AN000</td>
</tr>
<tr>
<td>V phase current measurement</td>
<td>PA02 / AN002</td>
<td>P001 / AN001</td>
<td>P001 / AN001</td>
</tr>
<tr>
<td>W phase current measurement</td>
<td>PA00 / AN000</td>
<td>P002 / AN002</td>
<td>P002 / AN002</td>
</tr>
<tr>
<td>PWM output (U_p)</td>
<td>PB04 / GTIOC4A</td>
<td>P409 / GTIOC1A</td>
<td>P409 / GTIOC1A</td>
</tr>
<tr>
<td>PWM output (V_p)</td>
<td>PB06 / GTIOC5A</td>
<td>P103 / GTIOC2A</td>
<td>P103 / GTIOC2A</td>
</tr>
<tr>
<td>PWM output (W_p)</td>
<td>PB08 / GTIOC6A</td>
<td>P111 / GTIOC3A</td>
<td>P111 / GTIOC3A</td>
</tr>
<tr>
<td>PWM output (U_n)</td>
<td>PB05 / GTIOC4B</td>
<td>P408 / GTIOC1B</td>
<td>P408 / GTIOC1B</td>
</tr>
<tr>
<td>PWM output (V_n)</td>
<td>PB07 / GTIOC5B</td>
<td>P102 / GTIOC2B</td>
<td>P102 / GTIOC2B</td>
</tr>
<tr>
<td>PWM output (W_n)</td>
<td>PB09 / GTIOC6B</td>
<td>P112 / GTIOC3B</td>
<td>P112 / GTIOC3B</td>
</tr>
<tr>
<td>Inductive sensor sine signal input</td>
<td>PC04 / AN014</td>
<td>P500 / AN016</td>
<td>P500 / AN016</td>
</tr>
<tr>
<td>Inductive sensor cosine signal input</td>
<td>PE15 / AN027</td>
<td>P006 / AN006</td>
<td>P006 / AN006</td>
</tr>
<tr>
<td>PWM emergency stop input at the</td>
<td>PC13 / GTETRGD</td>
<td>P104 / GTETRGB</td>
<td>P104 / GTETRGB</td>
</tr>
<tr>
<td>time of overcurrent detection</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
List of port interfaces of the sensor.

**Table 2-5 Port Interfaces**

<table>
<thead>
<tr>
<th>Function</th>
<th>MCI-LV-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>VCC</td>
<td>CN7 1pin</td>
</tr>
<tr>
<td>Inductive sensor sine signal input</td>
<td>CN7 6pin</td>
</tr>
<tr>
<td>Inductive sensor cosine signal input</td>
<td>CN7 8pin</td>
</tr>
<tr>
<td>GND</td>
<td>CN7 10pin</td>
</tr>
</tbody>
</table>
Vector control for permanent magnetic synchronous motor
with inductive sensor

2.2.4 Peripheral functions

List of the peripheral functions used in this system is given in Table 2-6.

Table 2-6 List of the Peripheral Functions

<table>
<thead>
<tr>
<th>Peripheral</th>
<th>Purpose</th>
<th>RA6T2</th>
<th>RA4T1</th>
<th>RA6T3</th>
</tr>
</thead>
<tbody>
<tr>
<td>12-bit A/D Converter</td>
<td>U phase current measurement</td>
<td>AN004</td>
<td>AN000</td>
<td>AN000</td>
</tr>
<tr>
<td></td>
<td>V phase current measurement</td>
<td>AN002</td>
<td>AN001</td>
<td>AN001</td>
</tr>
<tr>
<td></td>
<td>W phase current measurement</td>
<td>AN000</td>
<td>AN002</td>
<td>AN002</td>
</tr>
<tr>
<td></td>
<td>Inverter bus voltage measurement</td>
<td>AN006</td>
<td>AN004</td>
<td>AN004</td>
</tr>
<tr>
<td></td>
<td>For rotation speed command value input (analog value)</td>
<td>AN008</td>
<td>AN005</td>
<td>AN005</td>
</tr>
<tr>
<td></td>
<td>Inductive sensor sine signal input</td>
<td>AN014</td>
<td>AN016</td>
<td>AN016</td>
</tr>
<tr>
<td></td>
<td>Inductive sensor cosine signal input</td>
<td>AN027</td>
<td>AN006</td>
<td>AN006</td>
</tr>
<tr>
<td>AGT</td>
<td>Speed control interval timer</td>
<td>AGT0</td>
<td>AGT0</td>
<td>AGT0</td>
</tr>
<tr>
<td>GPT</td>
<td>U phase PWM output</td>
<td>CH4</td>
<td>CN1</td>
<td>CN1</td>
</tr>
<tr>
<td></td>
<td>V phase PWM output</td>
<td>CH5</td>
<td>CN2</td>
<td>CN2</td>
</tr>
<tr>
<td></td>
<td>W phase PWM output</td>
<td>CH6</td>
<td>CN3</td>
<td>CN3</td>
</tr>
<tr>
<td>POEG</td>
<td>PWM emergency stop input at the time of overcurrent detection</td>
<td>Group D</td>
<td>Group B</td>
<td>Group B</td>
</tr>
</tbody>
</table>

2.2.4.1 RA6T2

(1) 12-bit A/D Converter (12ADC)

U-phase current, V-phase current, W-phase current, inverter bus voltage, inductive sensor output and rotation speed command are measured in "Single scan mode" (use a hardware trigger).

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

The AGT is used as 500 [µs] interval timer.

(3) General PWM Timer (GPT)

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

(4) Port Output Enable for GPT (POEG)

The port executing PWM output are set to high impedance state when an overcurrent is detected (when a low level of the GTETRGD port is detected)
Vector control for permanent magnetic synchronous motor with inductive sensor

Figure 2-5 FSP Configuration of ADC Driver [1/4]
<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Channel 0</td>
<td></td>
</tr>
<tr>
<td>Scan Group</td>
<td>Scan Group 0</td>
</tr>
<tr>
<td>Channel Select</td>
<td>AN000</td>
</tr>
<tr>
<td>Sampling State Table ID</td>
<td>Sampling State Entry 0</td>
</tr>
<tr>
<td>Channel Gain Table</td>
<td>Disabled</td>
</tr>
<tr>
<td>Channel Offset Table</td>
<td>Disabled</td>
</tr>
<tr>
<td>Add/Average Mode</td>
<td>Disabled</td>
</tr>
<tr>
<td>Add/Average Count</td>
<td>1-time conversion (Normal Conversion)</td>
</tr>
<tr>
<td>Limit Clip Table Id</td>
<td>Disabled</td>
</tr>
<tr>
<td>Conversion Resolution Format Select</td>
<td>12-bit Data Format</td>
</tr>
<tr>
<td>Virtual Channel 1</td>
<td></td>
</tr>
<tr>
<td>Scan Group</td>
<td>Scan Group 0</td>
</tr>
<tr>
<td>Channel Select</td>
<td>AN000</td>
</tr>
<tr>
<td>Sampling State Table ID</td>
<td>Sampling State Entry 0</td>
</tr>
<tr>
<td>Channel Gain Table</td>
<td>Disabled</td>
</tr>
<tr>
<td>Channel Offset Table</td>
<td>Disabled</td>
</tr>
<tr>
<td>Add/Average Mode</td>
<td>Disabled</td>
</tr>
<tr>
<td>Add/Average Count</td>
<td>1-time conversion (Normal Conversion)</td>
</tr>
<tr>
<td>Limit Clip Table Id</td>
<td>Disabled</td>
</tr>
<tr>
<td>Conversion Resolution Format Select</td>
<td>12-bit Data Format</td>
</tr>
<tr>
<td>Virtual Channel 2</td>
<td></td>
</tr>
<tr>
<td>Scan Group</td>
<td>Scan Group 0</td>
</tr>
<tr>
<td>Channel Select</td>
<td>AN004</td>
</tr>
<tr>
<td>Sampling State Table ID</td>
<td>Sampling State Entry 0</td>
</tr>
<tr>
<td>Channel Gain Table</td>
<td>Disabled</td>
</tr>
<tr>
<td>Channel Offset Table</td>
<td>Disabled</td>
</tr>
<tr>
<td>Add/Average Mode</td>
<td>Disabled</td>
</tr>
<tr>
<td>Add/Average Count</td>
<td>1-time conversion (Normal Conversion)</td>
</tr>
<tr>
<td>Limit Clip Table Id</td>
<td>Disabled</td>
</tr>
<tr>
<td>Conversion Resolution Format Select</td>
<td>12-bit Data Format</td>
</tr>
<tr>
<td>Virtual Channel 3</td>
<td></td>
</tr>
<tr>
<td>Scan Group</td>
<td>Scan Group 1</td>
</tr>
<tr>
<td>Channel Select</td>
<td>AN006</td>
</tr>
<tr>
<td>Sampling State Table ID</td>
<td>Sampling State Entry 0</td>
</tr>
<tr>
<td>Channel Gain Table</td>
<td>Disabled</td>
</tr>
<tr>
<td>Channel Offset Table</td>
<td>Disabled</td>
</tr>
<tr>
<td>Add/Average Mode</td>
<td>Disabled</td>
</tr>
<tr>
<td>Add/Average Count</td>
<td>1-time conversion (Normal Conversion)</td>
</tr>
<tr>
<td>Limit Clip Table Id</td>
<td>Disabled</td>
</tr>
<tr>
<td>Conversion Resolution Format Select</td>
<td>12-bit Data Format</td>
</tr>
<tr>
<td>Virtual Channel 4</td>
<td></td>
</tr>
<tr>
<td>Scan Group</td>
<td>Scan Group 1</td>
</tr>
<tr>
<td>Channel Select</td>
<td>AN001</td>
</tr>
<tr>
<td>Sampling State Table ID</td>
<td>Sampling State Entry 0</td>
</tr>
<tr>
<td>Channel Gain Table</td>
<td>Disabled</td>
</tr>
<tr>
<td>Channel Offset Table</td>
<td>Disabled</td>
</tr>
<tr>
<td>Add/Average Mode</td>
<td>Disabled</td>
</tr>
<tr>
<td>Add/Average Count</td>
<td>1-time conversion (Normal Conversion)</td>
</tr>
<tr>
<td>Limit Clip Table Id</td>
<td>Disabled</td>
</tr>
</tbody>
</table>

Figure 2-6 FSP Configuration of ADC Driver [2/4]
Vector control for permanent magnetic synchronous motor with inductive sensor

Figure 2-7 FSP Configuration of ADC Driver [3/4]
Vector control for permanent magnetic synchronous motor with inductive sensor

Figure 2-8 FSP Configuration of ADC Driver [4/4]
<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter Checking</td>
<td>Default (BSP)</td>
</tr>
<tr>
<td>Pin Output Support</td>
<td>Disabled</td>
</tr>
<tr>
<td>Pin Input Support</td>
<td>Disabled</td>
</tr>
<tr>
<td>g_timer3 Timer, Low-Power (r_agt)</td>
<td></td>
</tr>
<tr>
<td>General</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>g_timer3</td>
</tr>
<tr>
<td>Channel</td>
<td>0</td>
</tr>
<tr>
<td>Mode</td>
<td>Periodic</td>
</tr>
<tr>
<td>Period</td>
<td>30000</td>
</tr>
<tr>
<td>Period Unit</td>
<td>Raw Counts</td>
</tr>
<tr>
<td>Count Source</td>
<td>PCLKB</td>
</tr>
<tr>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>Interrupts</td>
<td></td>
</tr>
<tr>
<td>Callback</td>
<td>rm_motor_speed_cyclic</td>
</tr>
<tr>
<td>Underflow Interrupt Priority</td>
<td>Priority 10</td>
</tr>
</tbody>
</table>

Figure 2-9 FSP Configuration of AGT Driver
### g_timer0 Timer, General PWM (r_gpt)

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common Parameter Checking</td>
<td>Default (BSP)</td>
</tr>
<tr>
<td>Pin Output Support</td>
<td>Enabled with Extra Features</td>
</tr>
<tr>
<td>Write Protect Enable</td>
<td>Disabled</td>
</tr>
<tr>
<td>Clock Source</td>
<td>PCLKD</td>
</tr>
</tbody>
</table>

#### Module g_timer0 Timer, General PWM (r_gpt)

##### General
- **Name**: g_timer0
- **Channel**: 4
- **Mode**: Triangle-Wave Symmetric PWM
- **Period**: 50
- **Period Unit**: Microseconds

##### Output
- **Custom Waveform**
  - **Duty Cycle Percent** (only applicable in PWM mode): 50
- **GTIOCA Output Enabled**: True
- **GTIOCA Stop Level**: Pin Level Low
- **GTIOCB Output Enabled**: True
- **GTIOCB Stop Level**: Pin Level High

##### Input

##### Interrupts

##### Extra Features
- **Output Disable**
  - **Output Disable POEG Trigger**
    - **POEG Link**: POEG Channel 3
    - **GTIOCA Disable Setting**: Set Hi Z
    - **GTIOCB Disable Setting**: Set Hi Z
- **ADC Trigger**
  - **Start Event Trigger (Channels with GTINTAD only)**
    - **Trigger Event A/D Converter Start Request A Durin**
    - **Trigger Event A/D Converter Start Request A Durin**
    - **Trigger Event A/D Converter Start Request B Durin**
    - **Trigger Event A/D Converter Start Request B Durin**
- **Dead Time**
  - **Dead Time Count Up (Raw Counts)**: 240
  - **Dead Time Count Down (Raw Counts)** (Channels with): 240
- **ADC Trigger (Channels with GTADTRA only)**
  - **ADC A Compare Match (Raw Counts)**: 0
- **ADC Trigger (Channels with GTADTRB only)**
  - **ADC B Compare Match (Raw Counts)**: 0
- **Interrupt Skipping (Channels with GTICT only)**
  - **Extra Features**: Enabled

---

Figure 2-10 FSP Configuration of GPT Driver
2.2.4.2 RA4T1

(1) 12-bit A/D Converter (12ADC)

U-phase current, V-phase current, W-phase current, inverter bus voltage, inductive sensor output and rotation speed command are measured in "Single scan mode" (use a hardware trigger).

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

The AGT is used as 500 [µs] interval timer.

(3) General PWM Timer (GPT)

On the channel 1, 2 and 3, output with dead time is performed by using the complementary PWM Output Operating Mode.

(4) Port Output Enable for GPT (POEG)

The port executing PWM output are set to high impedance state when an overcurrent is detected (when a low level of the GTETRGB port is detected)
Figure 2-12 Overall FSP Stacks diagram

Figure 2-13 FSP Configuration of ADC Driver [1/2]
Figure 2-14 FSP Configuration of ADC Driver [2/2]
Vector control for permanent magnetic synchronous motor with inductive sensor

Figure 2-15 FSP Configuration of AGT Driver
### Vector control for permanent magnetic synchronous motor with inductive sensor

#### Figure 2-16 FSP Configuration of GPT Driver

<table>
<thead>
<tr>
<th>g_timer0 Timer, General PWM (r.gpt)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Settings</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Common</strong></td>
<td></td>
</tr>
<tr>
<td>Parameter Checking</td>
<td>Default (BSP)</td>
</tr>
<tr>
<td>Pin Output Support</td>
<td>Enabled with Extra Features</td>
</tr>
<tr>
<td>Write Protect Enable</td>
<td>Disabled</td>
</tr>
<tr>
<td>Clock Source</td>
<td>PCLKD</td>
</tr>
<tr>
<td><strong>Module g_timer0 Timer, General PWM (r.gpt)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>General</strong></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>g_timer0</td>
</tr>
<tr>
<td>Channel</td>
<td>1</td>
</tr>
<tr>
<td>Mode</td>
<td>Triangle-Wave Symmetric PWM</td>
</tr>
<tr>
<td>Period</td>
<td>50</td>
</tr>
<tr>
<td>Period Unit</td>
<td>Microseconds</td>
</tr>
<tr>
<td><strong>Output</strong></td>
<td></td>
</tr>
<tr>
<td>Custom Waveform</td>
<td></td>
</tr>
<tr>
<td>Duty Cycle Percent (only applicable in PWM mode)</td>
<td>50</td>
</tr>
<tr>
<td>GTIOCA Output Enabled</td>
<td>True</td>
</tr>
<tr>
<td>GTIOCA Stop Level</td>
<td>Pin Level Low</td>
</tr>
<tr>
<td>GTIOCB Output Enabled</td>
<td>True</td>
</tr>
<tr>
<td>GTIOCB Stop Level</td>
<td>Pin Level High</td>
</tr>
<tr>
<td><strong>Input</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Interrupts</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Extra Features</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Output Disable</strong></td>
<td></td>
</tr>
<tr>
<td>Output Disable POEG Trigger</td>
<td></td>
</tr>
<tr>
<td>POEG Link</td>
<td>POEG Channel 1</td>
</tr>
<tr>
<td>GTIOCA Disable Setting</td>
<td>Level Low</td>
</tr>
<tr>
<td>GTIOCB Disable Setting</td>
<td>Level Low</td>
</tr>
<tr>
<td><strong>ADC Trigger</strong></td>
<td></td>
</tr>
<tr>
<td>Start Event Trigger (Channels with GTINTAD only)</td>
<td></td>
</tr>
<tr>
<td>Trigger Event A/D Converter Start Request A During Up Counting</td>
<td>□</td>
</tr>
<tr>
<td>Trigger Event A/D Converter Start Request A During Down Counting</td>
<td>□</td>
</tr>
<tr>
<td>Trigger Event A/D Converter Start Request B During Up Counting</td>
<td>□</td>
</tr>
<tr>
<td>Trigger Event A/D Converter Start Request B During Down Counting</td>
<td>□</td>
</tr>
<tr>
<td><strong>Dead Time</strong></td>
<td></td>
</tr>
<tr>
<td>Dead Time Count Up (Raw Counts)</td>
<td>200</td>
</tr>
<tr>
<td>Dead Time Count Down (Raw Counts) (Channels with GTDVD only)</td>
<td>200</td>
</tr>
<tr>
<td><strong>ADC Trigger</strong></td>
<td></td>
</tr>
<tr>
<td>(Channels with GTADTRA only)</td>
<td></td>
</tr>
<tr>
<td>ADC A Compare Match (Raw Counts)</td>
<td>0</td>
</tr>
<tr>
<td><strong>ADC Trigger</strong></td>
<td></td>
</tr>
<tr>
<td>(Channels with GTADTRB only)</td>
<td></td>
</tr>
<tr>
<td>ADC B Compare Match (Raw Counts)</td>
<td>0</td>
</tr>
<tr>
<td><strong>Interrupt Skipping</strong></td>
<td></td>
</tr>
<tr>
<td>(Channels with GTITC only)</td>
<td></td>
</tr>
<tr>
<td>Extra Features</td>
<td>Enabled</td>
</tr>
<tr>
<td><strong>Pins</strong></td>
<td></td>
</tr>
<tr>
<td>GTIOCA1</td>
<td>P409</td>
</tr>
<tr>
<td>GTIOCA1B</td>
<td>P408</td>
</tr>
</tbody>
</table>

---

For Renesas Flexible Motor Control Series
2.2.4.3 RA6T3

(1) 12-bit A/D Converter (12ADC)

U-phase current, V-phase current, W-phase current, inverter bus voltage, inductive sensor output and rotation speed command are measured in "Single scan mode" (use a hardware trigger).

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

The AGT is used as 500 [µs] interval timer.

(3) General PWM Timer (GPT)

On the channel 1, 2 and 3, output with dead time is performed by using the complementary PWM Output Operating Mode.

(4) Port Output Enable for GPT (POEG)

The port executing PWM output are set to high impedance state when an overcurrent is detected (when a low level of the GTETRGB port is detected)
## Vector control for permanent magnetic synchronous motor with inductive sensor

**Figure 2-19 FSP Configuration of ADC Driver [1/2]**

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Common</strong></td>
<td>[ ] Parameter Checking: Default (BSP)</td>
</tr>
<tr>
<td><strong>Module g_adc0 ADC (r_adc)</strong></td>
<td></td>
</tr>
<tr>
<td><strong>General</strong></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>g_adc0</td>
</tr>
<tr>
<td>Unit</td>
<td>0</td>
</tr>
<tr>
<td>Resolution</td>
<td>12-Bit</td>
</tr>
<tr>
<td>Alignment</td>
<td>Right</td>
</tr>
<tr>
<td>Clear after read</td>
<td>On</td>
</tr>
<tr>
<td>Mode</td>
<td>Single Scan</td>
</tr>
<tr>
<td>Double-trigger</td>
<td>Disabled</td>
</tr>
<tr>
<td><strong>Interrupts</strong></td>
<td></td>
</tr>
<tr>
<td>Normal/Group A Trigger</td>
<td>GPT1 COUNTER UNDERFLOW (Underflow)</td>
</tr>
<tr>
<td>Group B Trigger</td>
<td>Disabled</td>
</tr>
<tr>
<td>Group Priority (Valid only in Group Scan Mode)</td>
<td>Group A cannot interrupt Group B</td>
</tr>
<tr>
<td>Callback</td>
<td>rm_motor_driver_cyclic</td>
</tr>
<tr>
<td>Scan End Interrupt Priority</td>
<td>Priority 5</td>
</tr>
<tr>
<td>Scan End Group B Interrupt Priority</td>
<td>Disabled</td>
</tr>
<tr>
<td>Window Compare A Interrupt Priority</td>
<td>Disabled</td>
</tr>
<tr>
<td>Window Compare B Interrupt Priority</td>
<td>Disabled</td>
</tr>
</tbody>
</table>


### Vector control for permanent magnetic synchronous motor with inductive sensor

**Figure 2-20 FSP Configuration of ADC Driver [2/2]**

<table>
<thead>
<tr>
<th>g_adc0 ADC (r_adc)</th>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Settings</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>API Info</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Module g_adc0 ADC (r_adc)**
- **General**
- **Input**
  - Channel Scan Mask (channel availability varies by)
    - Channel 0
    - Channel 1
    - Channel 2
    - Channel 3
    - Channel 4
    - Channel 5
    - Channel 6
    - Channel 7
    - Channel 8
    - Channel 9
    - Channel 10
    - Channel 11
    - Channel 12
    - Channel 13
    - Channel 14
    - Channel 15
    - Channel 16
    - Channel 17
    - Channel 18
    - Channel 19
    - Channel 20
    - Channel 21
    - Channel 22
    - Channel 23
    - Channel 24
    - Channel 25
    - Channel 26
    - Channel 27
  - Temperature Sensor
  - Voltage Sensor
  - Group B Scan Mask (channel availability varies by)
  - Addition/Averaging Mask (channel availability varies by)
  - Sample and Hold
    - Sample and Hold Channels (Available only on)
      - Channel 0
      - Channel 1
      - Channel 2
    - Sample Hold States (Applies only to channels 0–24)
  - Window Compare
  - Add/Average Count
  - Reference Voltage control
    - VREFH0/VREFH

- **Extra**
<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter Checking</td>
<td>Default (BSP)</td>
</tr>
<tr>
<td>Pin Output Support</td>
<td>Disabled</td>
</tr>
<tr>
<td>Pin Input Support</td>
<td>Disabled</td>
</tr>
<tr>
<td>g_timer3 Timer, Low-Power (r_agt)</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>g_timer3</td>
</tr>
<tr>
<td>Channel</td>
<td>0</td>
</tr>
<tr>
<td>Mode</td>
<td>Periodic</td>
</tr>
<tr>
<td>Period</td>
<td>500</td>
</tr>
<tr>
<td>Period Unit</td>
<td>Microseconds</td>
</tr>
<tr>
<td>Count Source</td>
<td>PCLKB</td>
</tr>
<tr>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>Interrupts</td>
<td></td>
</tr>
<tr>
<td>Callback</td>
<td>rm_motor_speed_cyclic</td>
</tr>
<tr>
<td>Underflow Interrupt Priority</td>
<td>Priority 13</td>
</tr>
</tbody>
</table>
### Figure 2-22 FSP Configuration of GPT Driver

<table>
<thead>
<tr>
<th>Settings</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>API Info</td>
<td></td>
</tr>
<tr>
<td>Common</td>
<td></td>
</tr>
<tr>
<td>Parameter Checking</td>
<td>Default (BSP)</td>
</tr>
<tr>
<td>Pin Output Support</td>
<td>Enabled with Extra Features</td>
</tr>
<tr>
<td>Write Protect Enable</td>
<td>Disabled</td>
</tr>
<tr>
<td>Clock Source</td>
<td>PCLKD</td>
</tr>
<tr>
<td>Module g_timer0 Timer, General PWM (r.gpt)</td>
<td></td>
</tr>
<tr>
<td>General</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>g_timer0</td>
</tr>
<tr>
<td>Channel</td>
<td>1</td>
</tr>
<tr>
<td>Mode</td>
<td>Triangle-Wave Symmetric PWM</td>
</tr>
<tr>
<td>Period</td>
<td>50</td>
</tr>
<tr>
<td>Period Unit</td>
<td>Microseconds</td>
</tr>
<tr>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>Custom Waveform</td>
<td></td>
</tr>
<tr>
<td>Duty Cycle Percent (only applicable in PWM mode)</td>
<td>50</td>
</tr>
<tr>
<td>GTIOCA Output Enabled</td>
<td>True</td>
</tr>
<tr>
<td>GTIOCA Stop Level</td>
<td>Pin Level Low</td>
</tr>
<tr>
<td>GTIOCB Output Enabled</td>
<td>True</td>
</tr>
<tr>
<td>GTIOCB Stop Level</td>
<td>Pin Level High</td>
</tr>
<tr>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>Extra Features</td>
<td></td>
</tr>
<tr>
<td>Output Disable</td>
<td></td>
</tr>
<tr>
<td>POEG Link</td>
<td></td>
</tr>
<tr>
<td>POEG Channel 1</td>
<td></td>
</tr>
<tr>
<td>GTIOCA Disable Setting</td>
<td>Level Low</td>
</tr>
<tr>
<td>GTIOCB Disable Setting</td>
<td>Level Low</td>
</tr>
<tr>
<td>ADC Trigger</td>
<td></td>
</tr>
<tr>
<td>Start Event Trigger (Channels with GTINTAD only)</td>
<td></td>
</tr>
<tr>
<td>Trigger Event A/D Converter Start Request A During Up Counting</td>
<td></td>
</tr>
<tr>
<td>Trigger Event A/D Converter Start Request A During Down Counting</td>
<td></td>
</tr>
<tr>
<td>Trigger Event A/D Converter Start Request B During Up Counting</td>
<td></td>
</tr>
<tr>
<td>Trigger Event A/D Converter Start Request B During Down Counting</td>
<td></td>
</tr>
<tr>
<td>Dead Time</td>
<td></td>
</tr>
<tr>
<td>Dead Time Count Up (Raw Counts)</td>
<td>200</td>
</tr>
<tr>
<td>Dead Time Count Down (Raw Counts) (Channels with GTDVDD only)</td>
<td>200</td>
</tr>
<tr>
<td>ADC Trigger (Channels with GTADTRA only)</td>
<td></td>
</tr>
<tr>
<td>ADC ACompare Match (Raw Counts)</td>
<td>0</td>
</tr>
<tr>
<td>ADC Trigger (Channels with GTADTRB only)</td>
<td></td>
</tr>
<tr>
<td>ADC BCompare Match (Raw Counts)</td>
<td>0</td>
</tr>
<tr>
<td>Interrupt Skipping (Channels with GTITC only)</td>
<td></td>
</tr>
<tr>
<td>Extra Features</td>
<td>Enabled</td>
</tr>
<tr>
<td>Pins</td>
<td></td>
</tr>
<tr>
<td>GTIOC1A</td>
<td>P409</td>
</tr>
<tr>
<td>GTIOC1B</td>
<td>P408</td>
</tr>
</tbody>
</table>
Figure 2-23 FSP Configuration of POEG Driver
2.3 Software configuration

2.3.1 Software file configuration

Folder and file configuration of the software is given below.

Table 2-7  File and folder configuration[1/2]

<table>
<thead>
<tr>
<th>Folder</th>
<th>Subfolder</th>
<th>File</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>ra_cfg</td>
<td></td>
<td>ra_cfg</td>
<td>Generated config header</td>
</tr>
<tr>
<td>ra_gen</td>
<td></td>
<td>ra_gen</td>
<td>Generated register setting, main function etc</td>
</tr>
<tr>
<td>ra</td>
<td>arm</td>
<td>arm</td>
<td>CMSIS source code</td>
</tr>
<tr>
<td></td>
<td></td>
<td>board</td>
<td>Function definition for board</td>
</tr>
<tr>
<td></td>
<td></td>
<td>fsp/inc/api</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>bsp_api.h</td>
<td>BSP API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_adc_api.h</td>
<td>AD API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_elc_api.h</td>
<td>ELC API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_ioport_api.h</td>
<td>I/O API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_poeg_api.h</td>
<td>POEG API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_three_phase_api.h</td>
<td>3phase PWM API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_timer_api.h</td>
<td>Timer API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_transfer_api.h</td>
<td>Transfer API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_angle_api.h</td>
<td>Angle API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_api.h</td>
<td>Motor API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_current_api.h</td>
<td>Current API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_driver_api.h</td>
<td>Motor driver API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_inertia_estimate_api.h</td>
<td>Inertia estimate API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_inertia_estimate_api.h(Only RA4T1 and RA6T3)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_position_api.h</td>
<td>Position API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_return_origin_api.h(Only RA4T1 and RA6T3)</td>
<td>Return origin API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_speed_api.h</td>
<td>Speed API definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>fsp/inc/instances</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_adc_b.h(RA6T2) r_adc.h(RA4T1,RA6T3)</td>
<td>Function definition for AD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_agt.h</td>
<td>Function definition for AGT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_elc.h(Only RA4T1 and RA6T3)</td>
<td>Function definition for ELC</td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_gpt_three_phase.h</td>
<td>Function definition for 3 Phase PWM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_gpt.h</td>
<td>Function definition for GPT</td>
</tr>
</tbody>
</table>
### Table 2-8  File and folder configuration

<table>
<thead>
<tr>
<th>Folder</th>
<th>Subfolder</th>
<th>File</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>ra</td>
<td>fsp/inc/instances</td>
<td>r_ioport.h</td>
<td>Function definition for I/O</td>
</tr>
<tr>
<td></td>
<td></td>
<td>r_poeg.h</td>
<td>Function definition for POEG</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_current.h</td>
<td>Function definition for current control</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_driver.h</td>
<td>Function definition for motor driver</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_induction.h</td>
<td>Function definition for motor control</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_position_api.h</td>
<td>Function definition for position control</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_sense_induction.h</td>
<td>Function definition for angle/speed by inductive</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>sensors</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rm_motor_speed.h</td>
<td>Function definition for Speed</td>
</tr>
<tr>
<td>fsp/lib</td>
<td></td>
<td></td>
<td>Library files</td>
</tr>
<tr>
<td>fsp/src</td>
<td>bsp</td>
<td></td>
<td>BSP driver</td>
</tr>
<tr>
<td></td>
<td>r_adc_b/r_adc_b.c(RA6T2)</td>
<td></td>
<td>AD driver</td>
</tr>
<tr>
<td></td>
<td>r_adc/r_adc.c(RA4T1,RA6T3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>r_agt/r_agt.c</td>
<td></td>
<td>AGT driver</td>
</tr>
<tr>
<td></td>
<td>r_elc/r_elc.c(Only RA4T1 and RA6T3)</td>
<td></td>
<td>ELC driver</td>
</tr>
<tr>
<td></td>
<td>r_gpt/r_gpt.c</td>
<td></td>
<td>GPT driver</td>
</tr>
<tr>
<td></td>
<td>r_gpt_three_phase/ r_gpt_three_phase.c</td>
<td></td>
<td>3 phase PWM driver</td>
</tr>
<tr>
<td></td>
<td>r_ioport/r_ioport.c</td>
<td></td>
<td>I/O driver</td>
</tr>
<tr>
<td></td>
<td>r_poeg/r_poeg.c</td>
<td></td>
<td>POEG driver</td>
</tr>
<tr>
<td></td>
<td>rm_motor_current/rm_motor_current.c</td>
<td></td>
<td>Current control library driver</td>
</tr>
<tr>
<td></td>
<td>rm_motor_current/rm_motor_current_library.h</td>
<td></td>
<td>Current control library API definition</td>
</tr>
<tr>
<td></td>
<td>rm_motor_driver/rm_motor_driver.c</td>
<td></td>
<td>Motor driver</td>
</tr>
<tr>
<td></td>
<td>rm_motor_induction/rm_motor_induction.c</td>
<td></td>
<td>Motor control status driver</td>
</tr>
<tr>
<td></td>
<td>rm_motor_position/rm_motor_position.c</td>
<td></td>
<td>Position control driver</td>
</tr>
<tr>
<td></td>
<td>rm_motor_position/rm_motor_position_library.h</td>
<td></td>
<td>Position control library API definition</td>
</tr>
<tr>
<td></td>
<td>rm_motor_sense_induction/rm_motor_sense_induction.c</td>
<td></td>
<td>Angle detection with induction sensor driver</td>
</tr>
<tr>
<td></td>
<td>rm_motor_speed/rm_motor_speed.c</td>
<td></td>
<td>Speed control driver</td>
</tr>
<tr>
<td></td>
<td>rm_motor_speed/rm_motor_speed_library.h</td>
<td></td>
<td>Speed control library API definition</td>
</tr>
<tr>
<td>src</td>
<td>application/main</td>
<td>mtr_main.h, mtr_main.c</td>
<td>User main function</td>
</tr>
<tr>
<td></td>
<td></td>
<td>_mtr_control_parameter.h</td>
<td>Control parameters definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>_mtr_motor_parameter.h</td>
<td>Motor parameters definition</td>
</tr>
<tr>
<td></td>
<td>application/user/ics</td>
<td>r_mtr_ics.h, r_mtr_ics.c</td>
<td>Function definition for Analyzer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ICS2_RA6T2.h, ICS2_RA4T1.h, ICS2_RA6T3.h</td>
<td>Function definition for GUI tool</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ICS2_RA6T2.o, ICS2_RA4T1.o, ICS2_RA6T3.o</td>
<td>Communication library for GUI tool</td>
</tr>
</tbody>
</table>
2.3.2 Module configuration
Module configuration of the software is described below.

![Diagram of module configuration]

Figure 2-24 Module Configuration
2.4 Software specifications

Table 2-9 shows basic software specification of this system. For details of the vector control, refer to the application note ‘Vector control with encoder for permanent magnet synchronous motor (Algorithm)’ (R01AN3789).

Table 2-9 Basic Specifications of Vector Control with induction sensors Software

<table>
<thead>
<tr>
<th>Item</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control method</td>
<td>Vector control</td>
</tr>
<tr>
<td>Position detection method</td>
<td>Inductive sensor</td>
</tr>
<tr>
<td>Motor rotation start/stop</td>
<td>Determined depending on the level of SW1 or input from Renesas Motor Workbench</td>
</tr>
<tr>
<td>Input voltage</td>
<td>DC 24 [V]</td>
</tr>
<tr>
<td>Main clock frequency</td>
<td>RA6T2: 240 [MHz]</td>
</tr>
<tr>
<td></td>
<td>RA6T3: 200 [MHz]</td>
</tr>
<tr>
<td></td>
<td>RA4T1: 100 [MHz]</td>
</tr>
<tr>
<td>Carrier frequency (PWM)</td>
<td>20 [kHz] (Carrier period: 50 [µs])</td>
</tr>
<tr>
<td>Dead time</td>
<td>2 [µs]</td>
</tr>
<tr>
<td>Current control period</td>
<td>RA6T2: 50 [µs]</td>
</tr>
<tr>
<td></td>
<td>RA6T3: 50 [µs]</td>
</tr>
<tr>
<td></td>
<td>RA4T1: 100 [µs]</td>
</tr>
<tr>
<td>Speed control period</td>
<td>RA6T2: 500 [µs]</td>
</tr>
<tr>
<td></td>
<td>RA6T3: 500 [µs]</td>
</tr>
<tr>
<td></td>
<td>RA4T1: 1000 [µs]</td>
</tr>
<tr>
<td>Rotation speed control range</td>
<td>CW: 0 [rpm] to 4000 [rpm]</td>
</tr>
<tr>
<td></td>
<td>CCW: 0 [rpm] to 4000 [rpm]</td>
</tr>
<tr>
<td>Position control range</td>
<td>At board_ui</td>
</tr>
<tr>
<td></td>
<td>Position command generation: Voltage input divided by VR (input range) -180° to 180°</td>
</tr>
<tr>
<td></td>
<td>At ics_ui</td>
</tr>
<tr>
<td></td>
<td>Position command generation: Position profile of trapezoidal curve for speed command value (input range) -32768° to 32767° (Max speed) CW / CCW: 4000[rpm]</td>
</tr>
<tr>
<td>Optimization setting of compiler</td>
<td>Optimization level Optimize more(-O2) (default setting)</td>
</tr>
<tr>
<td>Processing stop for protection</td>
<td>Disables the motor 1 control signal output (six outputs), under any of the following conditions.</td>
</tr>
<tr>
<td></td>
<td>1. Instantaneous value of current of each phase exceeds 3.82(=1.8*sqrt(2)*1.5) [A] (RA6T2)</td>
</tr>
<tr>
<td></td>
<td>3.54(=1.67*sqrt(2)*1.5) [A] (RA4T1,RA6T3) (monitored in current control period)</td>
</tr>
<tr>
<td></td>
<td>2. Inverter bus voltage exceeds 60 [V] (monitored in current control period)</td>
</tr>
<tr>
<td></td>
<td>3. Inverter bus voltage is less than 8 [V] (monitored in current control period)</td>
</tr>
<tr>
<td></td>
<td>4. Rotation speed exceeds 4500 [rpm] (monitored in current control period)</td>
</tr>
<tr>
<td></td>
<td>When an external over current signal is detected (when a low level is detected), the PWM output ports are set to high impedance state.</td>
</tr>
</tbody>
</table>
2.5 Interrupt Priority

Table 2-10 shows the interrupt and priorities used in this system.

Table 2-10 Interrupt priority

<table>
<thead>
<tr>
<th>Interrupt level</th>
<th>Priority</th>
<th>function</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>AGT0 INT 500 [μs] Interrupt</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>ADC0 ADI0(RA6T2) ADC0 SCAN END(RA4T1,RA6T3) A/D complete interrupt</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>POEG3 EVENT(RA6T2) POEG1 EVENT(RA4T1,RA6T3) Over current error interrupt</td>
</tr>
<tr>
<td>0</td>
<td>Max</td>
<td></td>
</tr>
</tbody>
</table>

Figure 2-25 RA6T2 FSP Interrupts Configuration

Figure 2-26 RA4T1/RA6T3 FSP Interrupts Configuration
3. Descriptions of the control program

The target software of this application note is explained here.

3.1 Contents of control

3.1.1 Motor start/stop

The start and stop of the motor are controlled by input from Renesas Motor Workbench or SW1.

SW1 is assigned to a general-purpose port. When the port is at a “Low” level, it is determined that the start switch is being pressed. Conversely, when the level is switched to “High”, the software determines that the motor should be stopped.

3.1.2 A/D Converter

(1) Motor rotation speed reference

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

<table>
<thead>
<tr>
<th>Item</th>
<th>Conversion ratio (reference: A/D conversion value)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rotation speed</td>
<td></td>
</tr>
<tr>
<td>reference CW</td>
<td>0 rpm to 4000 rpm: 0800H to 0FFFH</td>
</tr>
<tr>
<td>CCW</td>
<td>0 rpm to 4000 rpm: 07FFH to 0000H</td>
</tr>
<tr>
<td>Position reference</td>
<td></td>
</tr>
<tr>
<td>CW</td>
<td>0 rpm to 180 degrees: 0800H to 0FFFH</td>
</tr>
<tr>
<td>CCW</td>
<td>0 rpm to 180 degrees: 07FFH to 0000H</td>
</tr>
</tbody>
</table>

(2) Inverter bus voltage

Inverter bus voltage is measured as given in Table 3-2. It is used for modulation factor calculation and over-voltage/low-voltage detection. (When an abnormality is detected, PWM is stopped.)

<table>
<thead>
<tr>
<th>Item</th>
<th>Conversion ratio (Inverter bus voltage: A/D conversion value)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inverter bus voltage</td>
<td>0 [V] to 73.26 [V]: 0000H to 0FFFH</td>
</tr>
</tbody>
</table>

(3) U, V, W phase current

The U, V and W phase currents are measured as shown in Table 3-3 and used for vector control. User can select only U and W phase currents to use as 2shunt resistances detection.

<table>
<thead>
<tr>
<th>Item</th>
<th>Conversion ratio (U, V, W phase current: A/D conversion value)</th>
</tr>
</thead>
<tbody>
<tr>
<td>U, V, W phase current</td>
<td>-8.25 [A] to 8.25 [A]: 0000H to 0FFFH (Note)</td>
</tr>
</tbody>
</table>

Current = (3.3V-1.65V)/(0.01Ohm * 20)=8.25A
3.1.3 Modulation (current control module)
A modulated voltage can be output to improve the efficiency of voltage usage. The modulation operation is set from the API of the current control module.

(a) Sine wave modulation (MOD_METHOD_SPWM)
The modulation factor \( m \) is defined as follows.

\[
m = \frac{V}{E}
\]

\( m \): Modulation ratio  \( V \): Reference voltage  \( E \): Inverter input voltage

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

\[
\begin{bmatrix}
V'_u \\
V'_v \\
V'_w
\end{bmatrix} = \begin{bmatrix}
V_u \\
V_v \\
V_w
\end{bmatrix} + \Delta V \begin{bmatrix}
1 \\
1 \\
1
\end{bmatrix}
\]

\[
\Delta V = -\frac{V_{\max} + V_{\min}}{2}, \quad V_{\max} = \max\{V_u, V_v, V_w\}, \quad V_{\min} = \min\{V_u, V_v, V_w\}
\]

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

\( V'_u, V'_v, V'_w \): Command values of U-, V-, and W-phases for PWM generation (modulation wave)

The modulation factor \( m \) is defined as follows.

\[
m = \frac{V'}{E}
\]

\( m \): Modulation ratio  \( V' \): Reference phase voltage for PWM  \( E \): Inverter input voltage
3.1.4 State transition

Figure 3-1 is a state transition diagram of the vector control with inductive sensors software. In the target software of this application note, the software state is managed by “SYSTEM MODE”.

(1). SYSTEM MODE

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

(2). EVENT

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

Table 3-4 List of EVENT

<table>
<thead>
<tr>
<th>EVENT name</th>
<th>occurrence factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>STOP</td>
<td>by user operation</td>
</tr>
<tr>
<td>RUN</td>
<td>by user operation</td>
</tr>
<tr>
<td>ERROR</td>
<td>when the system detects an error</td>
</tr>
<tr>
<td>RESET</td>
<td>by user operation</td>
</tr>
</tbody>
</table>
3.1.5 Start-up method

Figure 3-2 and Figure 3-3 show the software implementation of d-axis and inductive sensor alignment method. The d-axis alignment method used as startup control of position control method, in initialization mode (MOTOR_SENSE_INDUCTIVE_MODE_INIT) and Boot mode (MOTOR_SENSE_INDUCTIVE_MODE_BOOT). In drive mode (MOTOR_SENSE_INDUCTIVE_MODE_DRIVE) vector control is implemented for PMSM with inductive sensor. Each reference value setting of d-axis current, q-axis current and speed is managed by respective status.

Figure 3-2 Startup Speed Control of Vector Control PMSM with inductive sensor Software

Figure 3-3 Startup Position Control of Vector Control PMSM with inductive sensor Software
3.1.6 Generation of Position Profile
(Position profile of trapezoidal curve for speed command value)

In vector control software for PMSM with inductive 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 3-4 Generation of position profile]

Figure 3-4 Generation of position profile

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_max_speed_rpm)
- Position stabilization wait time (com_u2_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.
### 3.1.7 Speed measurement

The sample software calculates speed information from the deviation of angle information acquired at each current control cycle.

\[
\text{speed}[\text{rad/s}] = \frac{(\text{Position difference})}{(\text{Position get cycle})}
\]

Figure 3-5 Speed calculation using inductive sensor
### 3.1.8 Error correction of inductive sensor

This control program has a function to correct the analog output of the inductive 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 3-6 and the concept of phase correction is shown in Figure 3-7.

#### Figure 3-6 Analog output error gain correction concept

![Gain Correction Diagram](image)

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 3-8 shows the flow of data acquisition and correction.

In addition, regardless of whether the correction function is enabled or disabled, the d-axis alignment method is performed at startup, and the angle of the sensor detected at that time is corrected to 0 degrees.

#### Figure 3-7 Analog output error phase correction concept

![Phase Correction Diagram](image)
Vector control for permanent magnetic synchronous motor with inductive sensor

Calibration start

Initialize of calibration

Get gain correction data for open loop

Calculation of gain parameter

Get phase correction data for open loop

Calculation of phase parameter

Judge of calibration data

end

Figure 3-8 Error correction process flowchart
3.1.9 System protection function

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

- Over current error

The PWM output ports are set to high impedance state in response to an emergency stop signal (over current detection) from the hardware. In addition, U, V, and W phase currents are monitored in over current monitoring cycle. When an over current (when the current exceeds the over current limit) is detected, the CPU executes emergency stop (software detection). When this error occurs, the CPU performs emergency stop in the side of the motor in which the error occurred.

- Over voltage error

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

- Low voltage error

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

- 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. When this error occurs, the CPU performs emergency stop in the side of the motor in which the error occurred.

<table>
<thead>
<tr>
<th>Error name</th>
<th>Threshold</th>
<th>Monitoring cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over current error</td>
<td>Over current limit[A]</td>
<td>3.82(RA6T2), 3.54(RA4T1,RA6T3)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Current control period</td>
</tr>
<tr>
<td>Over voltage error</td>
<td>Over voltage limit[V]</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Current control period</td>
</tr>
<tr>
<td>Low voltage error</td>
<td>Low voltage limit[V]</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Current control period</td>
</tr>
<tr>
<td>Over speed error</td>
<td>Speed limit[rpm]</td>
<td>4500</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Current control period</td>
</tr>
</tbody>
</table>
3.1.10 AD triggers

Shows the timing of AD triggers and scan groups.

Figure 3-9 AD trigger timing

Figure 3-10 AD trigger timing (one time decimation)
3.2 Function specifications of vector control with inductive sensors software

The block diagram of the vector control with inductive sensor is shown below.

Figure 3-11 Block Diagram of Vector Control with inductive sensors
### Table 3-6 List of Functions Executed in Current Control Period Interrupt (1/4)

<table>
<thead>
<tr>
<th>File name</th>
<th>Function name</th>
<th>Process overview</th>
</tr>
</thead>
<tbody>
<tr>
<td>mtr_main.c</td>
<td>mtr_callback_event</td>
<td>Vector control with inductive sensor callback function</td>
</tr>
</tbody>
</table>
|                    | Input : (motor_callback_args_t *) p_args
|                    | / Callback argument                     |                                                                                 |
|                    | Output : None                           |                                                                                 |
| rm_motor_induction.c| rm_motor_induction_current_callback    | Set the speed control output to the current control input                        |
|                    | Input : (motor_current_callback_args_t *) p_args
|                    | / Callback argument                     |                                                                                 |
|                    | Output : None                           |                                                                                 |
|                    | RM_MOTOR_INDUCTION_ErrorCheck          | Check the occurrence of Error.                                                   |
|                    | Input : (motor_ctrl_t * const) p_ctrl / Pointer to control structure.  
|                    | (uint16_t * const) p_error / Pointer to get occured error
|                    | Output : fsp_err_t / Execution result |                                                                                 |
|                    | rm_motor_induction_copy_speed_current  | Copy speed output data to current input data                                     |
|                    | Input : (motor_speed_output_t *) st_output / Pointer to the structure of Speed Control output
|                    | (motor_current_input_t *) st_input / Pointer to the structure of Current Control input
|                    | Output : None                           |                                                                                 |
| rm_motor_driver.c  | rm_motor_driver_cyclic                 | Motor driver callback function                                                   |
|                    | Input : (adc_callback_args_t *) p_args
|                    | / Callback argument                     |                                                                                 |
|                    | Output : None                           |                                                                                 |
|                    | rm_motor_driver_current_get            | Get A/D converted data (Phase Current & Main Line Voltage)                      |
|                    | Input : (motor_driver_instance_ctrl_t *) p_ctrl / The pointer to the motor driver module instance
|                    | Output : None                           |                                                                                 |
|                    | RM_MOTOR_DRIVER_FlagCurrentOffsetGet   | Measure current offset values                                                    |
|                    | Input : (motor_driver_ctrl_t * const) p_ctrl / Pointer to control structure  
|                    | (uint8_t * const) p_flag_offset / Flag of finish current offset detection
|                    | Output : fsp_err_t / Execution result |                                                                                 |
|                    | RM_MOTOR_DRIVER_PhaseVoltageSet        | Set Phase Voltage Data to calculate PWM duty.                                    |
|                    | Input : (motor_driver_ctrl_t * const) p_ctrl / Pointer to control structure  
|                    | (float const) u_voltage / U phase voltage
|                    | (float const) v_voltage / V phase voltage
|                    | (float const) w_voltage / W phase voltage
|                    | Output : fsp_err_t / Execution result |                                                                                 |
|                    | rm_motor_driver_modulation             | Perform PWM modulation                                                           |
|                    | Input : (motor_driver_instance_ctrl_t *) p_ctrl / The pointer to the motor driver module instance
|                    | Output : None                           |                                                                                 |
### Table 3-7 List of Functions Executed in Current Control Period Interrupt (2/4)

<table>
<thead>
<tr>
<th>File name</th>
<th>Function name</th>
<th>Process overview</th>
</tr>
</thead>
<tbody>
<tr>
<td>rm_motor_driver.c</td>
<td>rm_motor_driver_mod_run</td>
<td>Calculates duty cycle from input 3-phase voltage (bipolar)</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_driver_instance_ctrl_t *) p_ctrl / Pointer to Motor Driver instance (const float *) p_f4_v_in / Pointer to the 3-phase input voltage (float *) p_f4_duty_out / Where to store the 3-phase output duty cycle Output : None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>rm_motor_driver_set_uvw_duty</td>
<td>PWM duty setting</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_driver_instance_ctrl_t *) p_ctrl / Pointer to Motor Driver instance (float) f_duty_u / The duty cycle of Phase-U (float) f_duty_v / The duty cycle of Phase-V (float) f_duty_w / The duty cycle of Phase-W Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RM_MOTOR_DRIVER_CurrentGet</td>
<td>Get calculated phase Current, Vdc &amp; Va_max data</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_driver_callback_args_t *) p_args / Callback argument Output : None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RM_MOTOR_CURRENT_ParameterSet</td>
<td>Set (Input) Parameter Data.</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_ctrl_t * const) p_ctrl / Pointer to control structure (motor_current_input_t const * const) p_st_input / Pointer to input data structure Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RM_MOTOR_CURRENT_CurrentSet</td>
<td>Set d/q-axis Current &amp; Voltage Data.</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_ctrl_t * const) p_ctrl / Pointer to control structure (motor_current_input_current_t const * const) p_st_current / Pointer to input current structure (motor_current_input_voltage_t const * const) p_st_voltage / Pointer to input voltage structure Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RM_MOTOR_CURRENT_CurrentGet</td>
<td>Get d/q-axis Current.</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_ctrl_t * const) p_ctrl / Pointer to control structure (float * const) p_id / Pointer to get d-axis current (float * const) p_iq / Pointer to get q-axis current Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
<tr>
<td></td>
<td>motor_current_transform_uvw_dq_abs</td>
<td>Coordinate transform UVW to dq (absolute transform)</td>
</tr>
<tr>
<td></td>
<td>Input : (const float) f_angle / rotor angle (const float *) f_uvw / the pointer to the UVW-phase array in [U,V,W] format (float *) f_dq / where to store the [d,q] formatted array on dq coordinates Output : None</td>
<td></td>
</tr>
</tbody>
</table>
### Table 3-8 List of Functions Executed in Current Control Period Interrupt (3/4)

<table>
<thead>
<tr>
<th>File name</th>
<th>Function name</th>
<th>Process overview</th>
</tr>
</thead>
<tbody>
<tr>
<td>rm_motor_current.c</td>
<td>motor_current_angle_cyclic</td>
<td>Angle/Speed Process in Cyclic Process of Current Control</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_instance_t * const) p_instance / The pointer to current control module control instance</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RM_MOTOR_CURRENT_SpeedPhaseSet</td>
<td>Set Current Speed &amp; rotor phase Data.</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_ctrl_t const) p_ctrl / Pointer to control structure (float const) speed / Rotational speed (float const) phase / Rotor phase</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RM_MOTOR_CURRENT_CurrentReferenceSet</td>
<td>Set Current Reference Data</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_ctrl_t const) p_ctrl / Pointer to control structure (float const) id_reference / D-axis current Reference (float const) iq_reference / Q-axis current Reference</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RM_MOTOR_CURRENT_PhaseVoltageGet</td>
<td>Gets the set phase voltage.</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_ctrl_t const) p_ctrl / Pointer to control structure (motor_current_get_voltage_t const) p_voltage / Pointer to get Voltages</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
<tr>
<td></td>
<td>motor_current_pi_calculation</td>
<td>Calculates the output voltage vector from current vector command and actual current vector</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_instance_ctrl_t * const) p_ctrl / The pointer to the FOC current control structure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>motor_current_pi_control</td>
<td>PI control</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_pi_params_t * const) pi_ctrl / The pointer to the PI control structure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : float / PI control output value</td>
<td></td>
</tr>
<tr>
<td></td>
<td>motor_current_limit_abs</td>
<td>Limit with absolute value</td>
</tr>
<tr>
<td></td>
<td>Input : (float) f4_value / Target value (float) f4_limit_value / Limit</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : float / Limited value</td>
<td></td>
</tr>
<tr>
<td></td>
<td>motor_current_decoupling</td>
<td>Decoupling control</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_instance_ctrl_t * const) p_ctrl / The pointer to the FOC current control instance (float) f_speed_rad / The electrical speed (const motor_current_motor_parameter_t * const) p_mtr / The pointer to the motor parameter data structure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>motor_current_voltage_limit</td>
<td>Limit voltage vector</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_instance_ctrl_t * const) p_ctrl / The pointer to the FOC current control structure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : None</td>
<td></td>
</tr>
</tbody>
</table>
Table 3-9 List of Functions Executed in Current Control Period Interrupt (4/4)

<table>
<thead>
<tr>
<th>File name</th>
<th>Function name</th>
<th>Process overview</th>
</tr>
</thead>
<tbody>
<tr>
<td>rm_motor_current.c</td>
<td>motor_current_transform_dq_uvw_abs</td>
<td>Coordinate transform dq to UVW 3-phase (absolute transform)</td>
</tr>
<tr>
<td></td>
<td>Input : (const float) f_angle / Rotor angle (const float *) f_dq / The pointer to the dq-axis value array in [D,Q] format (float *) f_uvw / Where to store the [U,V,W] formatted 3-phase quantities array Output : None</td>
<td></td>
</tr>
<tr>
<td>librm_motor_current.a</td>
<td>rm_motor_voltage_error_compensation_main</td>
<td>Voltage error compensation</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_voltage_compensation_t *) st_volt_comp / Voltage error compensation data (float *) p_f4_v_array / Reference voltage (float *) p_f4_i_array / Reference current (float) f4_vdc / Bus voltage Output : None</td>
<td></td>
</tr>
<tr>
<td>rm_motor_sense_induction.c</td>
<td>RM_MOTOR_SENSE_INDUCTION_FlagPiCtrlSet</td>
<td>Set the flag of PI Control runs.</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_angle_ctrl_t * const) p_ctrl / Pointer to control structure (uint32_t const) flag_pi / The flag of PI control runs Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RM_MOTOR_SENSE_INDUCTION_AngleSpeedGet</td>
<td>Gets the current rotor's angle and rotation speed. (phase error data is invalid.)</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_angle_ctrl_t * const) p_ctrl / Pointer to control structure (float * const) p_angle / Memory address to get rotor angle data (float * const) p_speed / Memory address to get rotational speed data (float * const) p_phase_err / Memory address to get phase(angle) error data Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
<tr>
<td>r_gpt_three_phase.c</td>
<td>R_GPT_THREE_PHASE_DutyCycleSet</td>
<td>Sets duty cycle for all three timers.</td>
</tr>
<tr>
<td></td>
<td>Input : (three_phase_ctrl_t * const) p_ctrl / Control block set in @ref three_phase_api_t::open call for this timer (three_phase_duty_cycle_t * const) p_duty_cycle / Duty cycle values for all three timer channels Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
</tbody>
</table>
Table 3-10 List of Functions Executed in Speed Control Interrupt (1/2)

<table>
<thead>
<tr>
<th>File name</th>
<th>Function name</th>
<th>Process overview</th>
</tr>
</thead>
<tbody>
<tr>
<td>mtr_callback_event</td>
<td>Input : (motor_callback_args_t *) p_args / Callback argument</td>
<td>Vector control with induction sensors callback function</td>
</tr>
<tr>
<td></td>
<td>Output : None</td>
<td></td>
</tr>
<tr>
<td>get_vr1</td>
<td>Input : None</td>
<td>Get VR1 A/D conversion value</td>
</tr>
<tr>
<td></td>
<td>Output : uint16_t / conversion value</td>
<td></td>
</tr>
<tr>
<td>RM_MOTOR_CURRENT_ParameterGet</td>
<td>Input : (motor_current_ctrl_t * const) p_ctrl / Pointer to control structure (motor_current_output_t * const) p_st_output / Pointer to output data structure</td>
<td>Get speed control input data from current control</td>
</tr>
<tr>
<td>rm_motor_current.c</td>
<td>Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
<tr>
<td>rm_motor_induction.c</td>
<td>rm_motor_induction_speed_callback</td>
<td>Speed control callback function</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_speed_callback_args_t *) p_args / Callback argument</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : None</td>
<td></td>
</tr>
<tr>
<td>rm_motor_induction.c</td>
<td>rm_motor_induction_copy_current_speed</td>
<td>Copy current output data to speed input data</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_current_output_t *) p_output / Pointer to the structure of Current Control output (motor_speed_input_t *) p_input / Pointer to the structure of Speed Control input</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : None</td>
<td></td>
</tr>
<tr>
<td>rm_motor_speed.c</td>
<td>rm_motor_speed_cyclic Input : (timer_callback_args_t *) p_args/ Callback argument</td>
<td>Cyclic process of Speed Control (Call at timer interrupt)</td>
</tr>
<tr>
<td></td>
<td>Output : None</td>
<td></td>
</tr>
<tr>
<td>RM_MOTOR_SPEED_ParameterSet</td>
<td>Input : (motor_speed_ctrl_t * const) p_ctrl / Pointer to control structure (motor_speed_input_t const * const) p_st_input / Pointer to structure to input parameters</td>
<td>Set speed Input parameters</td>
</tr>
<tr>
<td></td>
<td>Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
<tr>
<td>RM_MOTOR_SPEED_SpeedControl</td>
<td>Input : (motor_speed_ctrl_t * const) p_ctrl / Pointer to control structure</td>
<td>Calculates the d/q-axis current reference.(Main process of Speed Control)</td>
</tr>
<tr>
<td></td>
<td>Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
<tr>
<td>rm_motor_speed_set_speed_ref</td>
<td>Input : (motor_speed_instance_ctrl_t *) p_ctrl / The pointer to the FOC data instance</td>
<td>Updates the speed reference</td>
</tr>
<tr>
<td></td>
<td>Output : float / Speed reference</td>
<td></td>
</tr>
<tr>
<td>rm_motor_speed_set_iq_ref</td>
<td>Input : (motor_speed_instance_ctrl_t *) p_ctrl / The pointer to the ctrl instance</td>
<td>Updates the q-axis current reference</td>
</tr>
<tr>
<td></td>
<td>Output : float / iq reference</td>
<td></td>
</tr>
<tr>
<td>rm_motor_speed_set_id_ref</td>
<td>Input : (motor_speed_instance_ctrl_t *) p_ctrl / The pointer to the ctrl instance</td>
<td>Updates the d-axis current reference</td>
</tr>
<tr>
<td></td>
<td>Output : float / id reference</td>
<td></td>
</tr>
<tr>
<td>RM_MOTOR_SPEED_ParameterGet</td>
<td>Input : (motor_speed_ctrl_t * const) p_ctrl / The pointer to the ctrl instance (motor_speed_output_t * const) p_st_output / Pointer to get speed control parameters</td>
<td>Get speed control output parameters</td>
</tr>
<tr>
<td></td>
<td>Output : fsp_err_t / Execution result</td>
<td></td>
</tr>
</tbody>
</table>
Table 3-11 List of Functions Executed in Speed Control Interrupt (2/2)

<table>
<thead>
<tr>
<th>File name</th>
<th>Function name</th>
<th>Process overview</th>
</tr>
</thead>
<tbody>
<tr>
<td>librm_motor_speed.a</td>
<td>rm_motor_speed_first_order_lpf</td>
<td>First Order LPF</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_speed_lpf_t *) p_lpf / First order LPF structure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(float) f4_omega / Natural frequency</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(float) f4_ctrl_period / Control period</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>rm_motor_speed_fluxwnk_set_vamax</td>
<td>Sets the maximum magnitude of voltage vector</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_speed_flux_weakening_t *) p_fluxwkn / The pointer to flux weakening structure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(float) f4_va_max / maximum magnitude of voltage vector</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>rm_motor_speed_fluxwnk_run</td>
<td>Executes the flux-weakening module</td>
</tr>
<tr>
<td></td>
<td>Input : (motor_speed_flux_weakening_t *) p_fluxwkn / The pointer to flux weakening structure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(float) f4_speed_rad / The electrical speed of motor</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(const float *) p_f4_idq / The pointer to the measured current vector in format d/q</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(float *) p_f4_idq_ref / The pointer to the reference current vector in format d/q</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output : None</td>
<td></td>
</tr>
</tbody>
</table>
3.3 Contents of control
3.3.1 Configuration Options

The configuration options of the vector control with induction sensors module for motor can be configured using the RA Configurator. The changed options are automatically reflected to common_data.c/h and hal_data.c/h files when generating code. The option names and setting values are listed in the Table 3-12 shown as follows.

Table 3-12 Configuration Options for motor_induction module

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limit of over current (A)</td>
<td>When a phase current exceeds this value, PWM output ports are set to off.</td>
</tr>
<tr>
<td>Limit of over voltage (V)</td>
<td>When an inverter voltage exceeds this value, PWM output ports are set to off.</td>
</tr>
<tr>
<td>Limit of over speed (rpm)</td>
<td>When a rotation speed exceeds this value, PWM output ports are set to off.</td>
</tr>
<tr>
<td>Limit of over speed (rpm)</td>
<td>When an inverter voltage becomes below this value, PWM output ports are set to off.</td>
</tr>
</tbody>
</table>

Table 3-13 Configuration Options Initial Value(rm_motor_encoder.h)

<table>
<thead>
<tr>
<th>Options</th>
<th>RA6T2</th>
<th>RA4T1</th>
<th>RA6T3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limit of over current (A)</td>
<td>1.8</td>
<td>1.8</td>
<td>1.8</td>
</tr>
<tr>
<td>Limit of over voltage (V)</td>
<td>60.0</td>
<td>60.0</td>
<td>60.0</td>
</tr>
<tr>
<td>Limit of over speed (rpm)</td>
<td>4500.0</td>
<td>4500.0</td>
<td>4500.0</td>
</tr>
<tr>
<td>Limit of over speed (rpm)</td>
<td>8.0</td>
<td>8.0</td>
<td>8.0</td>
</tr>
</tbody>
</table>

3.3.2 Configuration Options for included modules

The vector control with induction sensors module includes below modules.

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

And also these included modules have each configuration parameters as same as the vector control with induction sensors module. The option names and setting values are listed in the tables shown as follows.
Table 3-14 Configuration Options for Current Control

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage error compensation</td>
<td>Selects whether to “enable” or “disable” voltage error compensation.</td>
</tr>
<tr>
<td>Shunt type</td>
<td>Selects how many shunt resistances to use current detection. Please set to “1shunt”.</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Pole pairs</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Resistance (ohm)</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Inductance of d-axis (H)</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Inductance of q-axis (H)</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Permanent magnetic flux (Wb)</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Rotor inertia (kgm^2)</td>
</tr>
<tr>
<td>Design Parameter</td>
<td>Current PI loop omega</td>
</tr>
<tr>
<td>Design Parameter</td>
<td>Current PI loop zeta</td>
</tr>
</tbody>
</table>

Table 3-15 Configuration Options Initial Value(rm_motor_current.h)

<table>
<thead>
<tr>
<th>Options</th>
<th>RA6T2</th>
<th>RA4T1</th>
<th>RA6T3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage error compensation</td>
<td>Enable</td>
<td>Enable</td>
<td>Enable</td>
</tr>
<tr>
<td>Shunt type</td>
<td>2shunt</td>
<td>2shunt</td>
<td>2shunt</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Pole pairs</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Resistance (ohm)</td>
<td>0.84</td>
<td>0.84</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Inductance of d-axis (H)</td>
<td>0.0011</td>
<td>0.0011</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Inductance of q-axis (H)</td>
<td>0.0011</td>
<td>0.0011</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Permanent magnetic flux (Wb)</td>
<td>0.00623</td>
<td>0.00623</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Rotor inertia (kgm^2)</td>
<td>0.0000041</td>
<td>0.0000041</td>
</tr>
<tr>
<td>Design Parameter</td>
<td>Current PI loop omega</td>
<td>300.0</td>
<td>300.0</td>
</tr>
<tr>
<td>Design Parameter</td>
<td>Current PI loop zeta</td>
<td>1.0</td>
<td>1.0</td>
</tr>
</tbody>
</table>
### Table 3-16 Configuration Options for Speed Control

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed control period (sec)</td>
<td>The period of speed control process [sec].</td>
</tr>
<tr>
<td>Step of speed climbing (rpm)</td>
<td>The step of speed fluctuation [rpm]. Program controls speed by this step at acceleration and deceleration.</td>
</tr>
<tr>
<td>Maximum rotational speed (rpm)</td>
<td>Maximum rotational speed [rpm]</td>
</tr>
<tr>
<td>Speed LPF omega</td>
<td>Speed LPF parameter omega [Hz].</td>
</tr>
<tr>
<td>Speed at Id climbing (rpm)</td>
<td>The threshold speed to control d-axis current increase [rad/s]. Program increases d-axis current at start up the motor rotation until the speed reaches this value.</td>
</tr>
<tr>
<td>Limit of q-axis current (A)</td>
<td>Limit of q-axis current [A].</td>
</tr>
<tr>
<td>Flux weakening</td>
<td>Select enable/disable of flux weakening control at high speed.</td>
</tr>
<tr>
<td>Design parameter</td>
<td>Speed PI loop omega</td>
</tr>
<tr>
<td>Design parameter</td>
<td>Speed PI loop zeta</td>
</tr>
<tr>
<td>Design parameter</td>
<td>Speed observer omega</td>
</tr>
<tr>
<td>Design parameter</td>
<td>Speed observer zeta</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Pole pairs</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Resistance (ohm)</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Inductance of d-axis (H)</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Inductance of q-axis (H)</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Permanent magnetic flux (Wb)</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Rotor inertia (kgm^2)</td>
</tr>
</tbody>
</table>

### Table 3-17 Configuration Options Initial Value (rm_motor_speed.h)

<table>
<thead>
<tr>
<th>Options</th>
<th>RA6T2</th>
<th>RA4T1</th>
<th>RA6T3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed control period (sec)</td>
<td>0.0005</td>
<td>0.0005</td>
<td>0.0005</td>
</tr>
<tr>
<td>Step of speed climbing (rpm)</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Maximum rotational speed (rpm)</td>
<td>4000</td>
<td>4000</td>
<td>4000</td>
</tr>
<tr>
<td>Speed LPF omega</td>
<td>10.0</td>
<td>10.0</td>
<td>10.0</td>
</tr>
<tr>
<td>Speed at Id climbing (rpm)</td>
<td>500</td>
<td>400</td>
<td>400</td>
</tr>
<tr>
<td>Limit of q-axis current (A)</td>
<td>1.8</td>
<td>1.8</td>
<td>1.8</td>
</tr>
<tr>
<td>Flux weakening</td>
<td>Disable</td>
<td>Disable</td>
<td>Disable</td>
</tr>
<tr>
<td>Design parameter</td>
<td>Speed PI loop omega</td>
<td>30.0</td>
<td>5.0</td>
</tr>
<tr>
<td>Design parameter</td>
<td>Speed PI loop zeta</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>Design parameter</td>
<td>Speed observer omega</td>
<td>200.0</td>
<td>200.0</td>
</tr>
<tr>
<td>Design parameter</td>
<td>Speed observer zeta</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Pole pairs</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Resistance (ohm)</td>
<td>0.84</td>
<td>0.84</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Inductance of d-axis (H)</td>
<td>0.0011</td>
<td>0.0011</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Inductance of q-axis (H)</td>
<td>0.0011</td>
<td>0.0011</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Permanent inductonic flux (Wb)</td>
<td>0.00623</td>
<td>0.00623</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Rotor inertia (kgm^2)</td>
<td>0.0000041</td>
<td>0.0000041</td>
</tr>
</tbody>
</table>
### Table 3-18 Configuration Options for Position control module

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position dead band</td>
<td>Position dead band</td>
</tr>
<tr>
<td>Position band limit</td>
<td>Zero position error range</td>
</tr>
<tr>
<td>Speed feedforward ratio</td>
<td>Speed feedforward ratio</td>
</tr>
<tr>
<td>Position omega</td>
<td>Position control omega parameter [Hz].</td>
</tr>
<tr>
<td>Period of speed control (sec)</td>
<td>Speed control execution cycle</td>
</tr>
<tr>
<td>Position Profiling</td>
<td>Interval time</td>
</tr>
<tr>
<td>Position Profiling</td>
<td>Accel time</td>
</tr>
<tr>
<td>Position Profiling</td>
<td>Maximum accel time</td>
</tr>
<tr>
<td>Position Profiling</td>
<td></td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Pole pairs</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Resistance (ohm)</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Inductance of d-axis (H)</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Inductance of q-axis (H)</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Permanent magnetic flux (Wb)</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Rotor inertia (kgm^2)</td>
</tr>
</tbody>
</table>

### Table 3-19 Configuration Options Initial Value(rm_motor_position.h)

<table>
<thead>
<tr>
<th>Options</th>
<th>RA6T2</th>
<th>RA4T1</th>
<th>RA6T3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position dead band</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Position band limit</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Speed feedforward ratio</td>
<td>0.8</td>
<td>0.8</td>
<td>0.8</td>
</tr>
<tr>
<td>Position omega</td>
<td>10.0</td>
<td>10.0</td>
<td>10.0</td>
</tr>
<tr>
<td>Period of speed control (sec)</td>
<td>0.0005</td>
<td>0.0005</td>
<td>0.0005</td>
</tr>
<tr>
<td>Position Profiling</td>
<td>Interval time</td>
<td>400</td>
<td>400</td>
</tr>
<tr>
<td>Position Profiling</td>
<td>Accel time</td>
<td>0.3</td>
<td>0.3</td>
</tr>
<tr>
<td>Position Profiling</td>
<td>Maximum accel time</td>
<td>11077.904</td>
<td>11077.904</td>
</tr>
<tr>
<td>Position Profiling</td>
<td>Acceleration maximum speed</td>
<td>4000.0</td>
<td>4000.0</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Pole pairs</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Resistance (ohm)</td>
<td>0.84</td>
<td>0.84</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Inductance of d-axis (H)</td>
<td>0.0011</td>
<td>0.0011</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Inductance of q-axis (H)</td>
<td>0.0011</td>
<td>0.0011</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Permanent magnetic flux (Wb)</td>
<td>0.00623</td>
<td>0.00623</td>
</tr>
<tr>
<td>Motor Parameter</td>
<td>Rotor inertia (kgm^2)</td>
<td>0.0000041</td>
<td>0.0000041</td>
</tr>
</tbody>
</table>
Table 3-20 Configuration Options for Angle and Speed with induction sensors

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency of current control (kHz)</td>
<td>PWM carrier frequency [kHz]</td>
</tr>
<tr>
<td>Decimation of Interrupt</td>
<td>Speed correction count</td>
</tr>
<tr>
<td>Counts to get signal</td>
<td>Error correction time division</td>
</tr>
<tr>
<td>Limit of signal error</td>
<td>Error judgment threshold at error correction</td>
</tr>
<tr>
<td>Coefficient of speed LPF</td>
<td>LPF factor</td>
</tr>
<tr>
<td>A/D reference voltage</td>
<td>A/D reference voltage [V]</td>
</tr>
<tr>
<td>A/D conversion scale</td>
<td>A/D conversion scale</td>
</tr>
<tr>
<td>Openloop speed (rpm)</td>
<td>Open loop rotation speed [rpm]</td>
</tr>
<tr>
<td>D-axis current at openloop (A)</td>
<td>Open loop d-axis current [A]</td>
</tr>
<tr>
<td>Angle adjustment times</td>
<td>Hold time setting at initialization of rotor angle</td>
</tr>
<tr>
<td>Induction sensor pole pairs</td>
<td>Inductive sensor pole pairs</td>
</tr>
<tr>
<td>Motor pole pairs</td>
<td>Motor pole pairs</td>
</tr>
</tbody>
</table>

Table 3-21 Configuration Options Initial Value(rm_motor_sense_induction.h)

<table>
<thead>
<tr>
<th>Options</th>
<th>RA6T2</th>
<th>RA4T1</th>
<th>RA6T3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency of current control (kHz)</td>
<td>20.0</td>
<td>10.0</td>
<td>10.0</td>
</tr>
<tr>
<td>Decimation of Interrupt</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Counts to get signal</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Limit of signal error</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Coefficient of speed LPF</td>
<td>0.07</td>
<td>0.07</td>
<td>0.07</td>
</tr>
<tr>
<td>A/D reference voltage</td>
<td>3.3</td>
<td>3.3</td>
<td>3.3</td>
</tr>
<tr>
<td>A/D conversion scale</td>
<td>4095.0</td>
<td>4095.0</td>
<td>4095.0</td>
</tr>
<tr>
<td>Openloop speed (rpm)</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
</tr>
<tr>
<td>D-axis current at openloop (A)</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>Angle adjustment times</td>
<td>512</td>
<td>512</td>
<td>512</td>
</tr>
<tr>
<td>Induction sensor pole pairs</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Motor pole pairs</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>
### Table 3-22 Configuration Options for Driver Access

<table>
<thead>
<tr>
<th>Configuration Options (rm_motor_driver.h)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Options</td>
<td></td>
</tr>
<tr>
<td>PWM timer frequency (MHz)</td>
<td>PWM Timer Clock Frequency [MHz]</td>
</tr>
<tr>
<td>PWM carrier period (micro seconds)</td>
<td>PWM Carrier Period [micro seconds]</td>
</tr>
<tr>
<td>Dead time (raw counts)</td>
<td>PWM Dead time [raw counts]</td>
</tr>
<tr>
<td>Current range (A)</td>
<td>Measurement Range of Electric current [A]</td>
</tr>
<tr>
<td>Voltage range (V)</td>
<td>Measurement Range of Inverter Voltage [V]</td>
</tr>
<tr>
<td>Counts for current offset measurement</td>
<td>Counts of measurement the offset of A/D Conversion at electric current input.</td>
</tr>
<tr>
<td>Shunt type</td>
<td>Selects how many shunt resistances to use current detection.</td>
</tr>
<tr>
<td>A/D conversion channel for U phase current</td>
<td>A/D channel for U-phase current</td>
</tr>
<tr>
<td>A/D conversion channel for V phase current</td>
<td>A/D channel for V-phase current</td>
</tr>
<tr>
<td></td>
<td>It is invalid at 2shunt detection.</td>
</tr>
<tr>
<td>A/D conversion channel for W phase current</td>
<td>A/D channel for W-phase current</td>
</tr>
<tr>
<td>A/D conversion channel for main line voltage</td>
<td>A/D channel for main line voltage</td>
</tr>
<tr>
<td>General</td>
<td>A/D conversion channel for sin signal</td>
</tr>
<tr>
<td>General</td>
<td>A/D conversion channel for cos signal</td>
</tr>
<tr>
<td>Input voltage</td>
<td>Range of input for main line voltage</td>
</tr>
<tr>
<td>Resolution of A/D conversion</td>
<td>Resolution of A/D conversion</td>
</tr>
<tr>
<td></td>
<td>Please set same value with ADC module setting.</td>
</tr>
<tr>
<td>Offset of A/D conversion for current</td>
<td>Offset level of A/D conversion input for current</td>
</tr>
<tr>
<td></td>
<td>Please set according to the circuit.</td>
</tr>
<tr>
<td>Conversion level of A/D conversion for voltage</td>
<td>Conversion level of A/D conversion for voltage</td>
</tr>
<tr>
<td></td>
<td>Please set when the CPU main voltage is different.</td>
</tr>
<tr>
<td>GTIOCA stop level</td>
<td>Output level of upper arm at stop status</td>
</tr>
<tr>
<td>GTIOCB stop level</td>
<td>Output level of lower arm at stop status</td>
</tr>
<tr>
<td>Maximum duty</td>
<td>Maximum duty of PWM</td>
</tr>
<tr>
<td></td>
<td>Maximum duty except dead time.</td>
</tr>
</tbody>
</table>
## Table 3-23 Configuration Options Initial Value (rm_motor_driver.h)

<table>
<thead>
<tr>
<th>Options</th>
<th>RA6T2</th>
<th>RA4T1</th>
<th>RA6T3</th>
</tr>
</thead>
<tbody>
<tr>
<td>PWM timer frequency (MHz)</td>
<td>120</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>PWM carrier period (micro seconds)</td>
<td>50</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>Dead time (raw counts)</td>
<td>240</td>
<td>200</td>
<td>200</td>
</tr>
<tr>
<td>Current range (A)</td>
<td>16.5</td>
<td>16.5</td>
<td>16.5</td>
</tr>
<tr>
<td>Voltage range (V)</td>
<td>73.26</td>
<td>73.26</td>
<td>73.26</td>
</tr>
<tr>
<td>Counts for current offset measurement</td>
<td>500</td>
<td>500</td>
<td>500</td>
</tr>
<tr>
<td>Shunt type</td>
<td>2shunt</td>
<td>2shunt</td>
<td>2shunt</td>
</tr>
<tr>
<td>A/D conversion channel for U phase current</td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>A/D conversion channel for V phase current</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>A/D conversion channel for W phase current</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>A/D conversion channel for main line voltage</td>
<td>6</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>General A/D conversion channel for sin signal</td>
<td>27</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>General A/D conversion channel for cos signal</td>
<td>28</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>Input voltage</td>
<td>24.0</td>
<td>24.0</td>
<td>24.0</td>
</tr>
<tr>
<td>Resolution of A/D conversion</td>
<td>0xFFF</td>
<td>0xFFF</td>
<td>0xFFF</td>
</tr>
<tr>
<td>Offset of A/D conversion for current</td>
<td>0x7FF</td>
<td>0x7FF</td>
<td>0x7FF</td>
</tr>
<tr>
<td>Conversion level of A/D conversion for voltage</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>GTIOCA stop level</td>
<td>Pin level Low</td>
<td>Pin level Low</td>
<td>Pin level Low</td>
</tr>
<tr>
<td>GTIOCB stop level</td>
<td>Pin level High</td>
<td>Pin level High</td>
<td>Pin level High</td>
</tr>
<tr>
<td>Maximum duty</td>
<td>0.9375</td>
<td>0.9375</td>
<td>0.9375</td>
</tr>
</tbody>
</table>
3.4 Control flowcharts

3.4.1 Main process

![Main Process Flowchart]

Figure 3-12 Main Process Flowchart
3.4.2 Current Control Period Interrupt (Carrier synchronized Interrupt) Process

![Flowchart of Current Control Period Interrupt (Carrier synchronized Interrupt) Process]

- Carrier synchronous interrupt
- Encoder counter processing
- Get U phase and W phase current values
- Get inverter bus voltage value.
- Error check
- Current offset detection
- Calculate V phase current
- SYSTEM MODE
- Offset adjustment
- Angle and speed calculation
- Angle and speed related Process
- (UVW)-(d-q) Transform
- Current PI control
- Decoupling control
- Voltage limit
- Phase-lead compensation
- (d-q)-(UVW) Transform
- Voltage error compensation
- PWM duty calculation
- PWM register setting

Figure 3-13 Current Control Period Interrupt (Carrier Interrupt) Process Flowchart
3.4.3 Speed Control Period Interrupt Process

Figure 3-14 Speed Control Period Interrupt Process Flowchart
3.4.4 Over Current Detection Interrupt Process

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

![Over Current Detection Interrupt Process Flowchart](image)

Figure 3-15 Over Current Detection Interrupt Process Flowchart
4. Project Operation Overview

4.1 Importing the Demo Project

The sample application provided with this document may be imported into e²studio using the steps in this section.

1. Select File → Import.

![Figure 4-1 File Menu](image)
2. Select “Existing Projects into Workspace”.

![Figure 4-2 Import Wizard Selection](image)

3. Click “Browse…” button and select the demo project. Click Finish button and the demo project is imported.

![Figure 4-3 Import Projects](image)

4.2 Building and Debugging

Refer to the "e²studio Getting Started Guide (R20UT4204)".
4.3 Quick Start

When executing the sample code only in the evaluation environment without using Renesas motor workbench, the Quick Start Sample Project can be executed with the following procedure.

1. After turning on stabilized power supply or executing reset, LED1, and LED2 on the inverter board are both off and the motor stops.

2. IF the toggle switch (SW1) on the inverter board is turned on, the motor starts to rotate. Every time the toggle switch (SW1) is changed, motor rotation starts/stops alternately. If the motor rotates normally, LED1 is on. However, if LED2 on the inverter board is also on, error is occurring.

3. In order to change the direction of the motor rotation, adjust it with the variable resistor (VR) on the inverter board.
   - Turn the variable resistor (VR) right: Motor rotates clockwise
   - Turn the variable resistor (VR) left: Motor rotates counterclockwise

4. If error occurs, LED2 on the inverter board lighten, and the motor rotation stops. To restore, the toggle switch (SW1) on the inverter board needs to be turned off, then the switch (SW2) to be pushed and released.

5. In order to stop the operation check, turn off the output of the stabilized power supply after making sure that the motor rotation has already stopped.
Vector control for permanent magnetic synchronous motor
with inductive sensor
For Renesas Flexible Motor Control Series

4.4 Motor Control Development Support Tool ‘Renesas Motor Workbench’

4.4.1 Overview

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

You can find ‘Renesas Motor Workbench’ on Renesas Electronics Corporation website.

---

**Figure 4-4 Renesas Motor Workbench – Appearance**

**Set up for “Renesas Motor Workbench”**

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

The following is an example of operating the motor using the Easy function.

4.4.2.1 Position control

- Set the control mode to Position
  
  (1) Select the Position tab.
  
  (2) Turn on “Position”.

  If it has already been turned on, turn it off and then turn it on again.

![Figure 4-5 Procedure for setting the control mode to Position](image)

- Change the user interface to use Renesas Motor Workbench
  
  (1) Turn on "RMW UI".

  If it has already been turned on, turn it off and then turn it on again.

![Figure 4-6 Procedure for changing to use Renesas Motor Workbench](image)

- Run the motor
  
  (1) Press the “Run” button
  
  (2) Enter the command position with the "Ref position" slider.

![Figure 4-7 Motor rotation procedure](image)
- Stop the motor
  (1) Press the “Stop” button

- Processing when it stops (error)
  (1) Turn on "Reset" button.
  (2) Turn off “Reset” button
4.4.2.2 Speed control

- Set the control mode to Speed
  
  (1) Select the Speed tab.

  (2) Turn on “Speed”.

  If it has already been turned on, turn it off and then turn it on again.

![Figure 4-10 Procedure for setting the control mode to Speed](image)

- Change the user interface to use Renesas Motor Workbench
  
  (1) Turn on "RMW UI".

  If it has already been turned on, turn it off and then turn it on again.

![Figure 4-11 Procedure for changing to use Renesas Motor Workbench](image)

- Run the motor
  
  (1) Press the “Run” button

  (2) Enter the command speed with the "Ref speed" slider.

![Figure 4-12 Motor rotation procedure](image)
- Stop the motor
  (1) Press the “Stop” button

- Processing when it stops (error)
  (1) Turn on "Reset“ button.
  (2) Turn off "Reset“ button
4.4.3 List of variables for Analyzer function

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

Table 4-1 List of Variables for Analyzer

<table>
<thead>
<tr>
<th>Variable name</th>
<th>Type</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>com_u1_sw_userif (*)</td>
<td>uint8_t</td>
<td>User interface switch</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0: GUI use 1: Board user interface use (default)</td>
</tr>
<tr>
<td>com_u1_mode_system (*)</td>
<td>uint8_t</td>
<td>State management 0: Stop mode 1: Run mode 3: Reset</td>
</tr>
<tr>
<td>com_u1_ctrl_loop_mode</td>
<td>uint8_t</td>
<td>Control loop mode switch</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0: Speed control 1: Position control (default)</td>
</tr>
<tr>
<td>com_f4_ref_speed_rpm</td>
<td>float</td>
<td>Speed reference (Mechanical) [rpm]</td>
</tr>
<tr>
<td>com_s2_ref_position_deg</td>
<td>int16_t</td>
<td>Position command value [degree]</td>
</tr>
<tr>
<td>com_u2_mtr_pp</td>
<td>uint16_t</td>
<td>Number of pole pairs</td>
</tr>
<tr>
<td>com_f4_mtr_r</td>
<td>float</td>
<td>Resistance [Ω]</td>
</tr>
<tr>
<td>com_f4_mtr_id</td>
<td>float</td>
<td>d-axis Inductance [H]</td>
</tr>
<tr>
<td>com_f4_mtr_lq</td>
<td>float</td>
<td>q-axis Inductance [H]</td>
</tr>
<tr>
<td>com_f4_mtr_m</td>
<td>float</td>
<td>Flux [Wb]</td>
</tr>
<tr>
<td>com_f4_mtr_j</td>
<td>float</td>
<td>Inertia [kgm^2]</td>
</tr>
<tr>
<td>com_f4_pos_omega</td>
<td>float</td>
<td>Natural frequency of current control system [Hz]</td>
</tr>
<tr>
<td>com_f4_sob_omega</td>
<td>float</td>
<td>Damping ratio of current control system</td>
</tr>
<tr>
<td>com_f4_sob_zeta</td>
<td>float</td>
<td>Natural frequency of speed control system [Hz]</td>
</tr>
<tr>
<td>com_f4_speed_omega</td>
<td>float</td>
<td>Damping ratio of speed control system</td>
</tr>
<tr>
<td>com_f4_speed_zeta</td>
<td>float</td>
<td>Natural frequency of the position loop [Hz]</td>
</tr>
<tr>
<td>com_f4_current_omega</td>
<td>float</td>
<td>Natural frequency of the speed observer [Hz]</td>
</tr>
<tr>
<td>com_f4_current_zeta</td>
<td>float</td>
<td>Damping ratio of the speed observer</td>
</tr>
<tr>
<td>com_f4_current_lq</td>
<td>float</td>
<td>d-axis current reference in open loop mode [A]</td>
</tr>
<tr>
<td>com_f4_id_up_time</td>
<td>float</td>
<td>d-axis current command value addition time [ms]</td>
</tr>
<tr>
<td>com_f4_max_speed_rpm</td>
<td>float</td>
<td>Maximum speed</td>
</tr>
<tr>
<td>com_f4_speed_limit_rpm</td>
<td>float</td>
<td>Over speed limit</td>
</tr>
<tr>
<td>com_u2_pos_dead_band</td>
<td>uint16_t</td>
<td>Dead band of position</td>
</tr>
<tr>
<td>com_u2_pos_band_limit</td>
<td>uint16_t</td>
<td>Positioning complete range</td>
</tr>
<tr>
<td>com_u2_interval_time</td>
<td>uint16_t</td>
<td>Time interval of the position command changes</td>
</tr>
<tr>
<td>com_f4_accel_time</td>
<td>float</td>
<td>Acceleration time [s] (for position control)</td>
</tr>
<tr>
<td>com_f4_speed_rate_limit</td>
<td>float</td>
<td>Acceleration limit [s] (for speed control)</td>
</tr>
</tbody>
</table>
4.4.4 Operation Example for Analyzer

Following example shows motor driving operation using Analyzer. Operation is using “Control Window” as shown in Figure 4-4. Regarding specification of “Control Window”, refer to ‘Renesas Motor Workbench User’s Manual’.

- Change the user interface to Analyzer

1. Confirm the check-boxes of column [W?] for ‘com_u1_sw_userif’ marks.
2. Input ‘0’ in the [Write] box of ‘com_u1_sw_userif’.
3. Click the ‘Write’ button.

- Driving the motor

1. The [W?] check boxes contain checkmarks for “com_u1_mode_system1”, “com_s2_ref_position_deg”, “com_u1_enable_write”.
2. Type a reference position value in the [Write] box of “com_s2_ref_position_deg”.
3. Click the “Write” button.
4. Click the “Read” button. Confirm the [Read] box of “com_s2_ref_position_deg”, “g_u1_enable_write”.
5. Enter the same value of “g_u1_enable_write” in the [Write] box of “com_u1_enable_write”.
6. Enter “1” in the [Write] box of “com_u1_mode_system”.
7. Click the “Write” button.

- Stop the motor

1. Enter “0” in the [Write] box of “com_u1_mode_system”.
2. Click the “Write” button.
- Error cancel operation

1. Enter “3” in the [Write] box of “com_u1_mode_system”.
2. Click the “Write” button.

4.4.5 Tuner function

To use the Tuner function, use the executable file provided by Renesas Motor Workbench or "RA6T2_MCILV1_SPM_IS_FOC_TUNER_E2S_V100" included in the sample software.

For details on how to use the Tuner function, refer to the Tuner function manual (R20AN0528) included in the Renesas Motor Workbench download file.
4.4.6 Example of changing communication speed

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

- Change the communication speed setting of the sample software (when the required communication rate is 10 Mbps)
  (1) Change the value of ICS_BRR in r_mtr_ics.h to 1.
  (2) Change the value of MTR_ICS_DECIMATION in r_mtr_ics.h to 1.

![Figure 4-18 Modification of r_mtr_ics.h](image)

- Change the communication speed setting of Renesas Motor Workbench to connect
  (1) Press the Clock button on the Main Window to change the value to 80,000,000
  This value was calculated by multiplying the default 8,000,000 by 10 because the UART communication baud rate was changed from 1Mbps to 10Mbps.
  (2) Select the COM of the connected kit in the COM of Connection

![Figure 4-19 Clock frequency setting](image)

If the connection fails, repeat the procedure for reconnecting after resetting the communication board.
4.4.7 How to use the built-in communication library

The procedure for connecting to Renesas Motor Workbench using the built-in communication library without using the communication board with the sample software is shown below.

- Connection between PC and CPU board
  (1) Connect the CPU board and PC via a USB / serial conversion board, etc.

- Preparing a project for built-in communication (example of RA6T2 921600bps)
  (1) Cancel the registration of ICS2_RA6T2.o

![Figure 4-20 Unregister ICS2_RA6T2.o](image)
(2) Register ICS2_RA6T2_Built_in.o

![Register ICS2_RA6T2.o](image)

Figure 4-21 Register ICS2_RA6T2.o

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

```
#define USE_BUILT_IN (1)
#elif USE_BUILT_IN
#define MTR_ICS_DECIMATION (1)

/* For ICS */
#define ICS_BAR
#define ICS_INT_MODE
```

![Modification of r_mtr_ics.h](image)

Figure 4-22 Modification of r_mtr_ics.h
- Change the communication baud rate setting of Renesas Motor Workbench to connect
  (1) Change the value to 921,600 with Baud rate Dialog from the Option menu of the Main Window.
  (2) Select the COM port of the connected kit in the COM of Connection.

![Figure 4-23 Baud rate setting](image-url)
5. Reference Documents

RA6T2 Group User’s Manual: Hardware (R01UH0951)
RA4T1 Group User’s Manual: Hardware (R01UH0998)
RA6T3 Group User’s Manual: Hardware (R01UH0999)
RA Flexible Software Package Documentation
Application note: ‘Encoder vector control for permanent magnet synchronous motor (Algorithm)’ (R01AN3789)
Renesas Motor Workbench User’s Manual (R21UZ0004)
Renesas Motor Workbench Quick start guide (R21QS0011)
MCK-RA6T2 User’s Manual (R12UZ0091)
MCK-RA4T1 User’s Manual (R12UZ0114)
MCK-RA6T3 User’s Manual (R12UZ0115)
Revision History

<table>
<thead>
<tr>
<th>Rev.</th>
<th>Date</th>
<th>Description</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>May 23, 2023</td>
<td>-</td>
<td>First edition issued</td>
</tr>
</tbody>
</table>
General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

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

1. Precaution against Electrostatic Discharge (ESD)

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

2. Processing at power-on

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

3. Input of signal during power-off state

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

4. Handling of unused pins

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

5. Clock signals

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

6. Voltage application waveform at input pin

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

7. Prohibition of access to reserved addresses

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

8. Differences between products

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

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

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

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

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

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

6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.

   - “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   - “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.

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

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

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

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

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

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

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

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

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

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

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/

Trademarks
Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

© 2023 Renesas Electronics Corporation. All rights reserved.