R7F0C807

BLDC Motor Control

R01AN2004EC0100
Rev.1.00
Sep 30, 2014

Introduction

This application note describes the sample program for operating the 3-phase brushless DC motor with hall sensor by R7F0C807.

Target Device

R7F0C807

When applying the sample program covered in this application note to another microcomputer with the same SFR (Special Function Register), modify the program according to the specifications for the target microcomputer and conduct an extensive evaluation of the modified program.

Contents

1. Specifications

This application note describes an example of outputing 6 channels PWM to control the 3-phase brushless DC motor by using RTO (real-time output controller) function module of R7F0C807.

Table 1.1 lists the peripheral functions to be used and their applications.

Table 1.1 Peripheral Functions and Their Applications

Peripheral Function	Use
TAU00, TAU01	PWM generation
TAU02	Interval timer
TAU03	Timer for 1ms, speed PI control per 5ms
P00/RTIO00 P01/RTIO01 P02/RTIO02 P03/RTIO03 P04/RTIO04 P05/RTIO05	PWM output (6 channels)
P137/INTP0	The forced cut-off trigger signal input (INTP0)
P11/INTP1 P15/INTP2 P14/INTP3	Hall signal input (hall a, hall b, hall c)
A/D converter	Setting of rotation speed and current detection
P13	Start/Stop push switch
P10	Rotation direction control

Refer to "Figure 3.1 Hardware Configuration" in "3. Description of the Hardware".

2. Operating Conditions

The sample code contained in this application note has been tested under the conditions below.

Table 2.1 Operation Confirmation Conditions

Item	Contents
MCU used	R7F0C807
Operating frequency	• High-speed on-chip oscillator clock (f_{HOCO}) : 20 MHz (typ.) • CPU/peripheral hardware clock (f_{CLK}): 20 MHz
Operating voltage	5.0 V (Operation enabled from 4.5 to 5.5 V) SPOR detection operation (V_{SPOR}): rising edge 4.28V (typ.), falling edge 4.00V (min.)
Integrated development environment	Renesas Electronics Corporation CubeSuite + V2.01.00
C compiler	Renesas Electronics Corporation CA78K0R V1.60

3. Description of the Hardware

3.1 Hardware Configuration Example

Figure 3.1 shows an example of hardware configuration that is used for this application note.

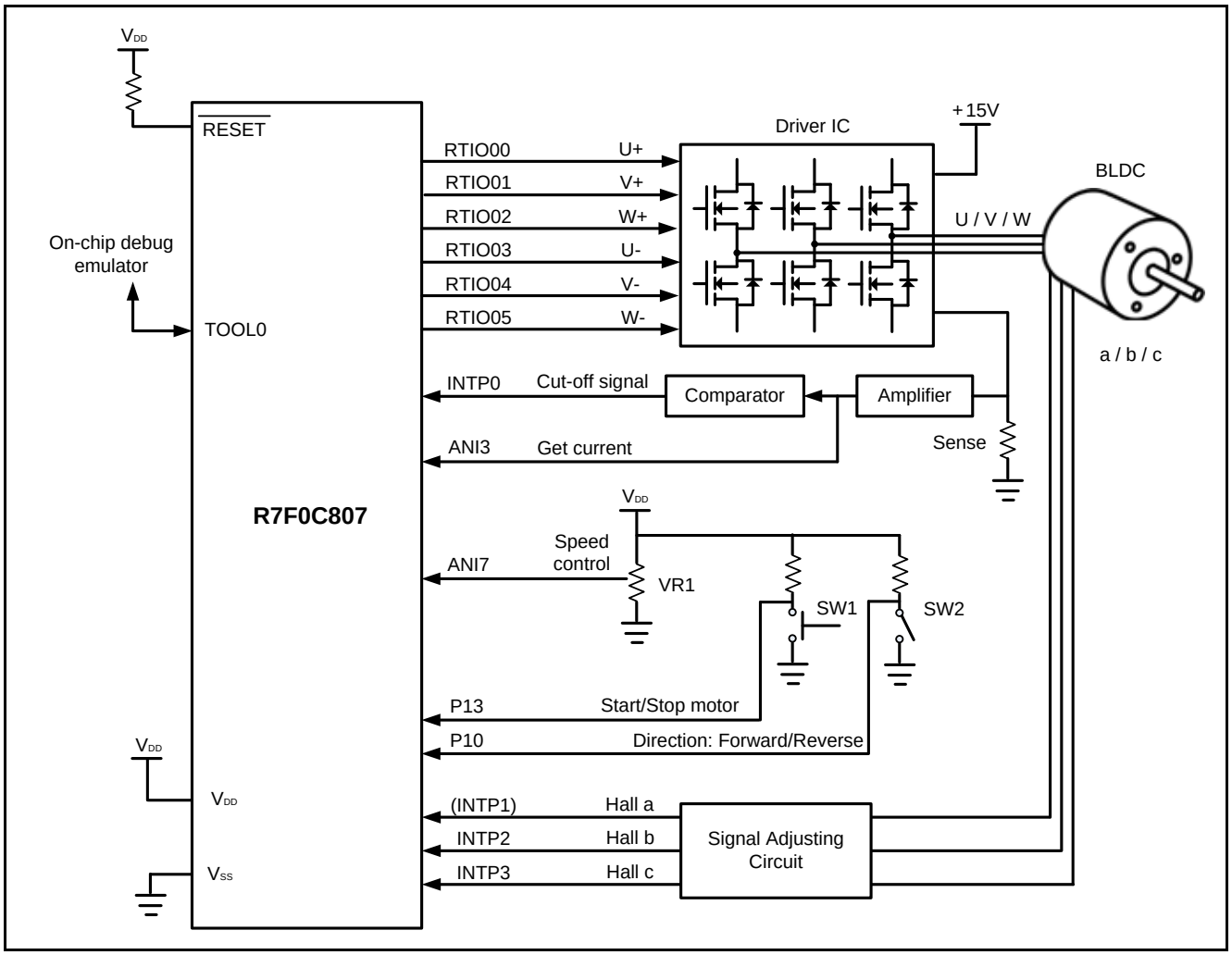

Figure 3.1 Hardware Configuration

Notes: 1. The purpose of this circuit is only to provide the connection outline and the circuit is simplified accordingly. When designing and implementing an actual circuit, provide proper pin treatment and make sure that the hardware's electrical specifications are met (connect the input-only ports separately to V_{DD} or V_{SS} via a resistor).

2. V_{DD} must be held at not lower than the reset release voltage (V_{SPOR}) that is specified as SPOR.

3.2 List of Pins to be Used

Table 3.1 lists the pins to be used and their function.

Table 3.1 Pins to be Used and Their Function

Pin Name	I/O	Description
RTIO00	Output	PWM output (U+)
RTIO01	Output	PWM output (V+)
RTIO02	Output	PWM output (W+)
RTIO03	Output	PWM output (U-)
RTIO04	Output	PWM output (V-)
RTIO05	Output	PWM output (W-)
INTP0	Input	Trigger signal for the function of forced cut-off
(INTP1)	Input	Hall signal input (Hall a)
INTP2	Input	Hall signal input (Hall b)
INTP3	Input	Hall signal input (Hall c)
ANI7	Input	Rotation speed command value input (analog value)
ANI3	Input	Sample the voltage of sense for current measurement
P13	Input	Start/Stop push switch
P10	Input	Input signal of motor rotation direction

3.3 Peripheral Functions

Table 3.2 lists the peripheral functions.

Table 3.2 Peripheral Functions List

Peripheral function	Use
Timer Array Unit (TAU0)	• PWM generation • Free run timer for rotation speed calculation • 1 ms interval timer
Real-time output controller (RTO)	• Output PWM to drive motor (six channels)
External interrupt (INTP0, INTP1, INTP2, INTP3)	• Input for the forced cut-off trigger signal • Input for Hall sensor signal (position detection)
A/D converter (ANI3, ANI7)	• The sense voltage measurement • Rotation speed command value input
I/O ports (P10, P13)	• Rotation direction input signal • Start/Stop input signal

(1) Timer array unit (TAU0)

(a) PWM generation (TAU00, TAU01)

PWM generation uses "PWM output function" of Timer Array Unit TAU0, channel 0 (master) and channel 1 (slave). PWM output from channel 1 as the input source of RTO.

(b) Free-run timer (TAU02)

Free-run timer for speed measurement uses "Interval timer function" of Timer Array Unit TAU02. However, it does not use the interrupt function.

(c) 1 ms interval timer (TAU03)

1 ms interval timer uses "Interval timer function"of Timer Array Unit TAU03. It uses the interrupt function.

Table 3.3 Timer Array Unit Usage Channel

No.	Use
Channel 0	PWM generation (master)
Channel 1	PWM generation (slave)
Channel 2	Free-run timer for speed measurement
Channel 3	1 ms interval timer

(2) Real-time output (RTO)

Real-time output (RTO) uses Timer Array Unit TAU0 channel 1 as input source, then controls PWM from TAU01 (such as inverting or not, high-level output , low-level output or Hi-z output) to drive the motor.

Table 3.4 Combination of Motor Control Signal Output and Real-Timer Output Pins

Real-timer output pins	Motor control signal
RTIO00	U+
RTIO01	V+
RTIO02	W+
RTIO03	U-
RTIO04	V-
RTIO05	W-

(3) Interrupts

External interrupt INTP0 is used as overcurrent detection, interrupt will be triggered when the motor current is over-limit, then RTO output low-level to cut off motor running to protect motor.

External interrupt INTP1, INTP2, INTP3 are used as position detection by hall sensor, input signals of hall sensor into 3 external interrupt ports and change the conduction pattern to keep motor running.

Table 3.5 Interrupts Usage List

Interrupt name	Interrupt source
INTP0	At the time of overcurrent detection (falling edge)
INTP1	Detects a variation of hall sensor signal a (both edges)
INTP2	Detects a variation of hall sensor signal b (both edges)
INTP3	Detects a variation of hall sensor signal c (both edges)
INTTM03	1 ms interval interrupt

(4) A/D converter

The rotation speed command value input and the sense voltage are measured by using A/D converter.

Resolution of A/D conversion is 10-bit, conversion speed is 3.4 us per channel and the smallest unit of conversion input value is given in Table 3.6.

Table 3.6 A/D Converter Correspondence Table

Item	Control value for A/D converter (1 bit)	Channel
Rotation speed command	2500 [rpm] / 1024 = 2.44 [rpm] (When A/D control value is 0, the minimum speed: 500 [rpm]) Speed range: 500 [rpm] ~ 3000 [rpm]	ANI7
The sense voltage	15 [V] / 1024 = 0.0146 [V]	ANI3

4. Motor Control Method

120° conducting control of BLDC motor with hall sensor and speed PI control are described below.

4.1 120° Conducting Control of the BLDC Motor with Hall Sensor

In this system, the hall sensor is used to detect the position of permanent magnet, then signals from the hall sensor signals are input to R7F0C807 as position information.

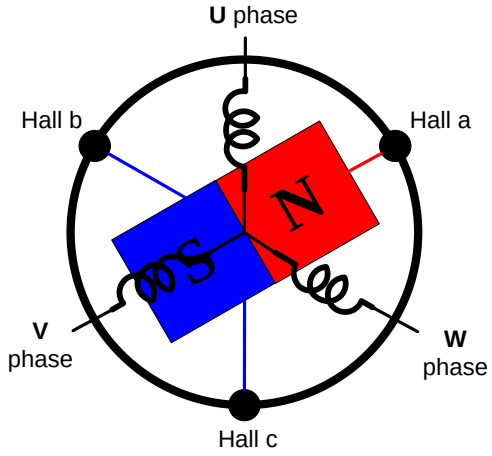

Figure 4.1 Example of Hall Sensor Position and Position Signal

As shown in Figure 4.1, a hall sensor is allocated every 120° and the respective hall sensor signals are switched depending on direction of rotating magnetic poles. Position information can be obtained every 60° (six patterns for one cycle) by combining these three hall sensor signals.

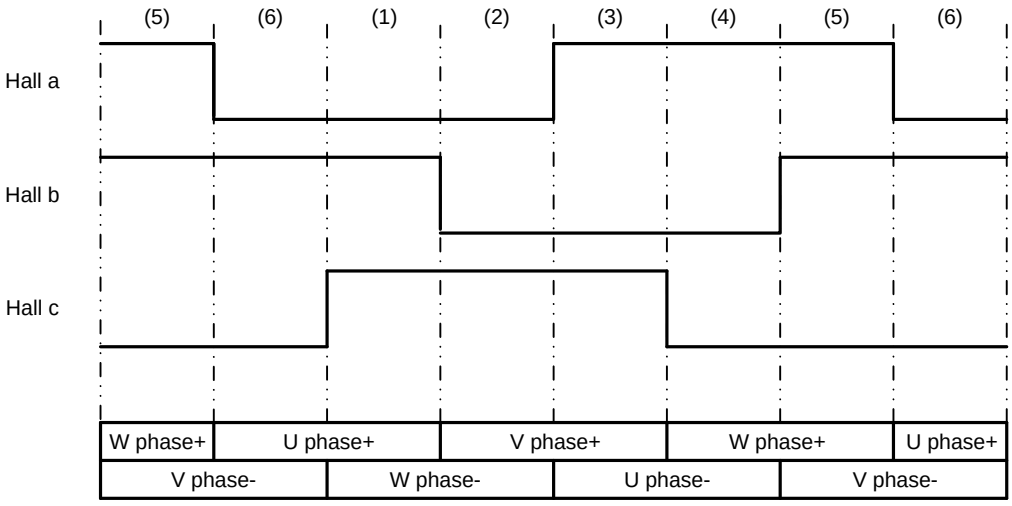

Figure 4.2 Relation between Hall Sensor Signals and Conduction Patterns

If the conduction patterns of each phase are changed in the switching timing of these hall sensor signals, as shown in Figure 4.2, rotating flux are generated as shown in Figure 4.3. Then the rotor has the torque and rotates.

As conduction duration of each switching element is 120°, this control method is referred to as 120° conducting control.

The relation between above-mentioned six conduction patterns and rotor position ranges is shown in Figure 4.3.

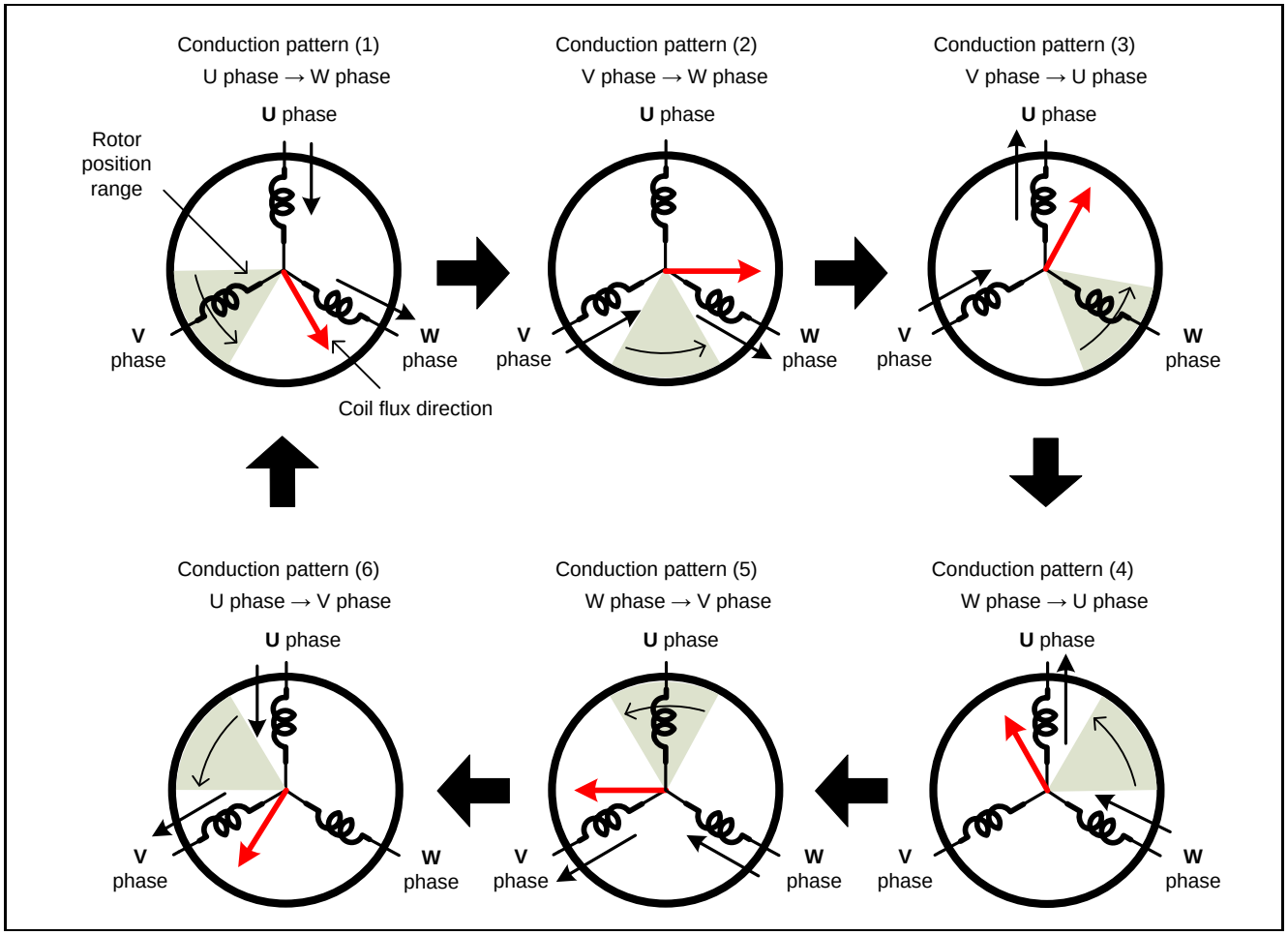

Figure 4.3 Six Conduction Patterns and Rotor Position Ranges

Supplements: 1. The relation between hall sensor signals and conduction patterns shown in Figure 4.3 is set to be suitable for this system. A different motor specification requires setting different conduction patterns appropriate to the system.
2. In the 120° conducting control, only six types of conduction patterns are generated for one cycle and hence in principle, a torque ripple occurs without fail.

4.2 Speed PI Control

In this system, the motor rotation speed is calculated from a difference of the present timer value and the timer value 2π[rad] before. Timer values are obtained through the external interrupt routine by hall sensor signals while having the timer of channel 2 of timer array unit operating in free running mode. This method is applicable even if three hall sensors are not placed at equal spaces.

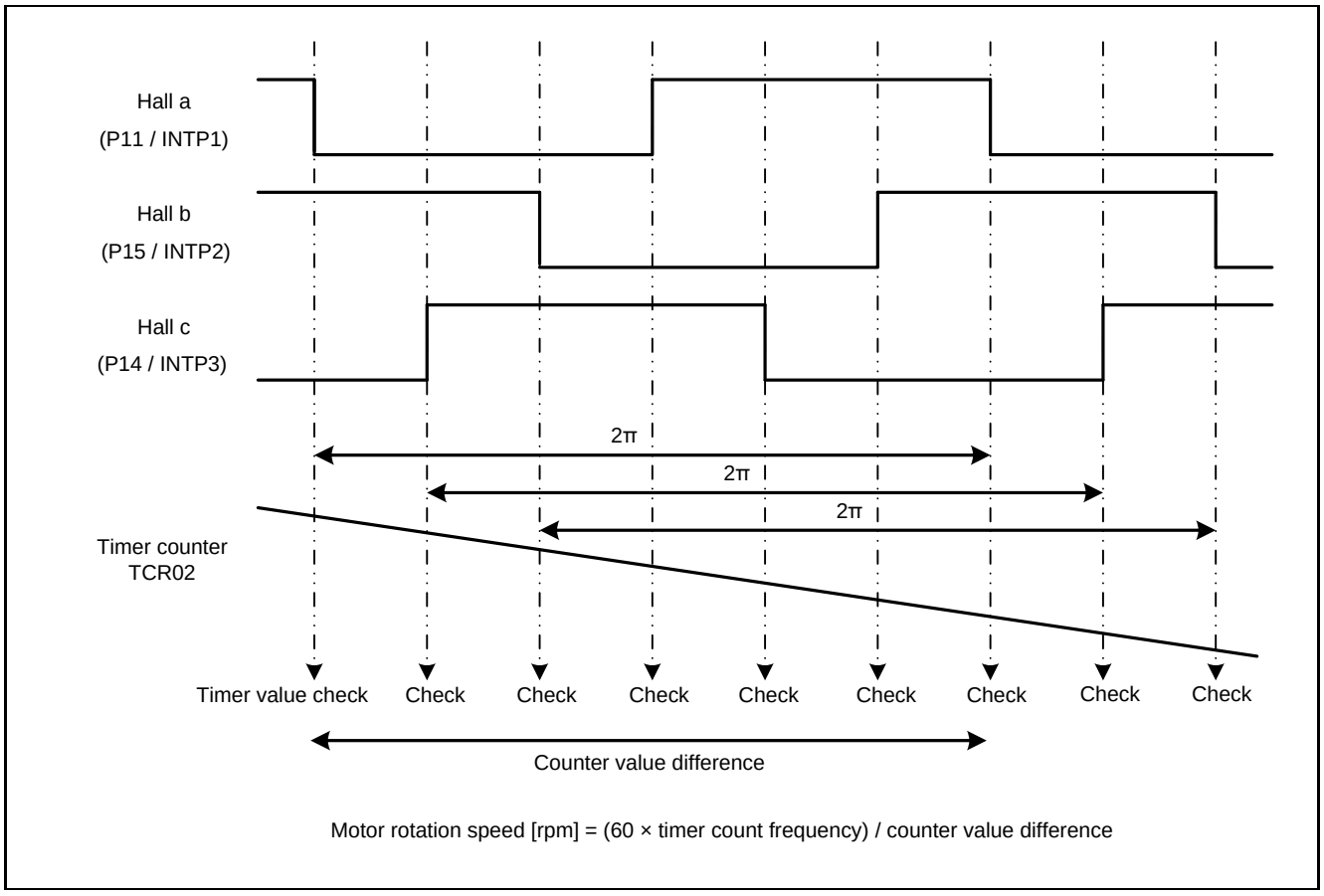

Figure 4.4 Method of Calculating Motor Rotation Speed

This system uses PI control for speed control. A duty command value at any (discrete) time n is calculated by the following formula.

$$\Delta duty = KP \times (err[n] - err[n-1]) + KI \times err[n]$$

Δduty: Duty err: Deviation of rotation speed command value and rotation speed calculation value
KP: Proportional gain KI: Integral gain

In this system, for starting motor and getting the rotor position, first 60° chopping is adopted, then change the conduction pattern constantly when the external interrupt is triggered. An example of motor control signal output waveforms at the time of first 60° chopping is given in Figure 4.5.

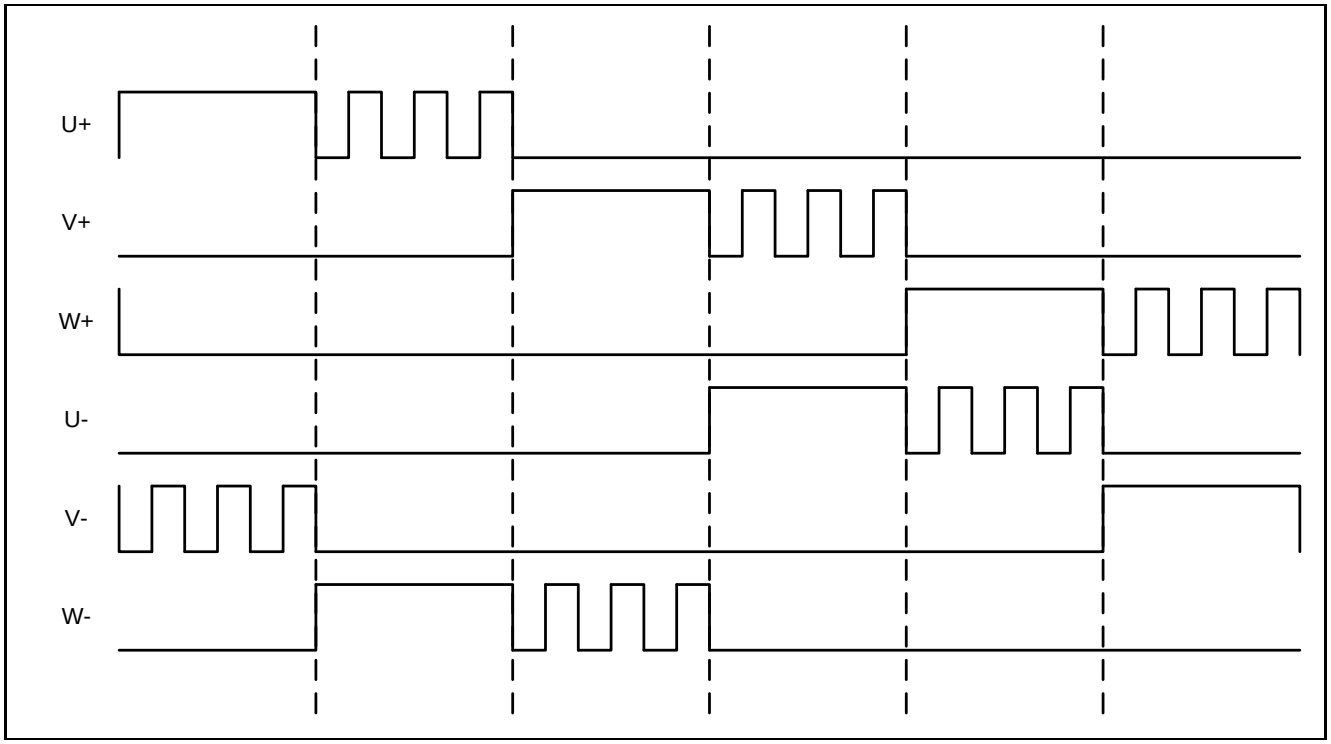

Figure 4.5 Motor startup Timing

5. Description of the Software

5.1 Operation Overview

In this application note, R7F0C807 uses 3 interrupt ports to acquire hall sensor signals. 6 channels RTO output conduction pattern to drive the motor. An interrupt routine controls the conduction pattern to keep motor running by changing 6 channels RTO output, which use hall sensor signals as interrupt trigger source. INTP0 is used as the forced cut-off signal input port and 6 channels RTO will output predefined cut-off level to stop motor when INTP0 is triggered by external signal.

(1) System initialization: Initialize Ports, Timer Array Unit (TAU), Real-time output (RTO), external interrupt and A/D converter.

(2) Start/Stop motor: If the Start/Stop key (SW1) is pressed at motor stop state, Start motor with speed 500rpm by PWM form RTO. Control the conduction pattern in interrupt routine, which use hall sensor signals as interrupt trigger source, and enable the measurement of rotation speed to calculate current speed. Press SW1 again, motor will stop running.

(3) Current detection: After starting motor, motor current (the user can output current information as required) will be detected by A/D port ANI3 that gets sense voltage value (amplifier is be used).

(4) Speed control: Adjust VR1 rotary knob, Target speed can be set by A/D port ANI7 through sampling VR1 voltage value. At program part, speed PI control will be proceeded per 5ms to make motor actual current speed and target speed be the same by adjusting the duty of PWM signal that output from 6 channels RTO.

(5) Overcurrent protection: If motor current exceeds the reference value in the process of motor rotation, interrupt INTP0 will be triggered and RTO outputs forced cut-off signal at the same time to stop the running motor immediately for safety. The forced cut-off state will be released after restarting motor.

5.2 List of Option Byte Settings

Table 5.1 summarizes the settings of the option bytes.

Table 5.1 Option Byte Settings

Address	Value	Description
000C0H	11101110B	Watchdog timer operation is stopped.(Count is stopped after reset.)
000C1H	11110011B	SPOR detection voltage: rising edge 4.28V(typ.), falling edge 4.00V(min.) P125/KR1/RESET pin: RESET input
000C2H	11111001B	HOCO: 20 MHz
000C3H	10000101B	On-chip debugging is enabled.

5.3 List of Constants

Table 5.2 lists the constants that used in this sample program.

Table 5.2 Constants for the Sample Program

Constant	Setting	Description
PCLK	20000000	PCLK (20MHz)
CARRIER	15000	Carrier cycle (15KHz)
PWM_INI_DUTY	165	The value of initial duty cycle (500rpm)
PWM_INI_PERIOD	PCLK/CARRIER	The value of PWM cycle (15KHz)
TAU02_FREQ	156000	TAU02 counter frequency (Hz)
KP_CONST	8	Proportional gain: 8 = (0.00008) * KP_KI_RATIO
KI_CONST	120	Integral gain: 120 = (0.0012) * KP_KI_RATIO
KP_KI_RATIO	100000	Amplification factor for proportional gain and integral gain (avoid teraflops)
MAX_REV	3000	Rotation speed command maximum value [rpm]
MIN_REV	500	Rotation speed command minimum value [rpm]
MAX_DUTY	385	Duty cycle maximum value (3000rpm)
NOW	0	Current state
PAST1	1	Previous state
ADC_VR1	7	ANI port connects with VR1 (ANI7)
ADC_VSENSE	3	ANI port connects with sense (ANI3)
VCC_REF	5000	Reference voltage for A/D converter (5000mV)
R_SENSE	510	The sense value (0.051Ω)
SPEED_OVERSIZE_LIMIT	144	Check whether speed value (uint16_t type) is overflow
PORT_START_STOP	P13	Input port (Start/Stop)
PORT_DIRECTION	P10	Input port (Forward/Reverse)
RTOOUTC0_Table[]	(0x10,0x01, 0x20,0x82, 0x48,0x04)	Forward: Positively switch 6 kinds of conduction patterns Reverse: Reversely switch 6 kinds of conduction patterns (Switch RTOOUTC0_Table[] and RTOOUTC1_Table[] at the same time)
RTOOUTC1_Table[]	(0x01,0x20, 0x02,0x00, 0x00,0x10)	Forward: Positively switch 6 kinds of conduction patterns Reverse: Reversely switch 6 kinds of conduction patterns (Switch RTOOUTC0_Table[] and RTOOUTC1_Table[] at the same time)

5.4 List of Variables

Table 5.3 lists the global variables that are used in this sample program.

Table 5.3 Constants for the Sample Program

Type	Variable Name	Contents	Function Used
uint8_t	g_motor_run_flag	Motor running state flag	mtr_tau03_interrupt() mtr_start_motor() mtr_stop_motor()
uint8_t	g_shutdown_flag	Motor forced cut-off flag	main() mtr_start_motor() mtr_over_current_interrupt()
uint8_t	g_pi_flag	PI control flag	main() mtr_tau03_interrupt() mtr_pi_ctrl_speed()
uint8_t	g_1msec_timer	1ms timer counter	mtr_tau03_interrupt() Delay_1ms()
int16_t	g_duty	Current value of duty cycle	mtr_start_motor() mtr_pi_ctrl_speed() TAU0_PWM_Duty()
uint16_t	g_speed_current	Current rotation speed	mtr_speed_calc() mtr_pi_ctrl_speed()

5.5 List of Functions

Table 5.4 summarizes the functions that are used in this sample program.

Table 5.4 Functions

Function Name	Outline
System_Ini	Initialization of system modules
PORT_Ini	Initialization of I/O ports
TAU0_Ini	Initialization of timer array unit (TAU0)
RTO_Ini	Initialization of real-time output (RTO)
INTP_Ini	Initialization of external interrupt (INTP)
AD_Ini	Initialization of A/D converter
main	Main function
TAU0_PWM_Duty	Adjust duty cycle function
Delay_1ms	Delay n×1ms function
mtr_start_motor	Motor start function
mtr_stop_motor	Motor stop function
mtr_current_detect	Current detection function
mtr_over_current_interrupt	Forced cut-off interrupt processing
mtr_hall_a_interrupt	External interrupt INTP1 processing
mtr_hall_b_interrupt	External interrupt INTP2 processing
mtr_hall_c_interrupt	External interrupt INTP3 processing
mtr_tau03_interrupt	1ms timer interrupt processing
mtr_speed_calc	Motor rotation speed calculation function
mtr_pi_ctrl_speed	Motor speed PI control function
mtr_get_adc	Execute A/D conversion function
mtr_eliminate_buffeting	Eliminate key buffeting function

5.6 Function Specifications

This section describes the specifications for the functions that are used in this sample program.

[Function Name] System_Ini

Synopsis	Initialization of system modules
Header	main.h
Declaration	void System_Ini(void)
Explanation	Initialization of Ports, Timer Array Unit (TAU), Real-time output (RTO), external interrupt and A/D converter.
Arguments	None
Return value	None
Remarks	None

[Function Name] PORT_Ini

Synopsis	Initialization of Ports
Header	main.h
Declaration	void PORT_Ini(void)
Explanation	Initialization of Ports state.
Arguments	None
Return value	None
Remarks	None

[Function Name] TAU0_Ini

Synopsis	Initialization of TAU0
Header	tau0.h
Declaration	void TAU0_Ini(void)
Explanation	Set TAU00 and TAU01 as PWM output mode, TAU02 as interval timer mode (Free-run timer), TAU03 as interval timer mode (1ms timer).
Arguments	None
Return value	None
Remarks	None

[Function Name] RTO_Ini

Synopsis	Initialization of RTO
Header	rto.h
Declaration	void RTO_Ini(void)
Explanation	Set PWM output from TAU01 as the input source for RTIO00~RTIO05, and enable the forced cut-off function.
Arguments	None
Return value	None
Remarks	None

[Function Name] INTP_Ini

Synopsis	Initialization of INTP
Header	intp.h
Declaration	void INTP_Ini(void)
Explanation	Set INTP0 valid edge: falling edge, INTP1,2,3 valid edge: both rising and falling edges.
Arguments	None
Return value	None
Remarks	None

[Function Name] AD_Ini

Synopsis	Initialization of A/D converter
Header	ad.h
Declaration	void AD_Ini(void)
Explanation	Selection of the A/D conversion resolution: 10 bit.
Arguments	None
Return value	None
Remarks	None

[Function Name] main

Synopsis	main function
Header	main.h
Declaration	void main(void)
Explanation	Motor Start/Stop key control, current detection, etc.
Arguments	None
Return value	None
Remarks	None

[Function Name] TAU0_PWM_Duty

Synopsis	Adjustment of duty cycle
Header	tau0.h, pi.h
Declaration	void TAU0_Ini(void)
Explanation	Adjustment of the regster value of duty cycle from TAU01.
Arguments	None
Return value	None
Remarks	None

[Function Name] Delay_1ms

Synopsis	Delay
Header	pi.h
Declaration	void Delay_1ms(uint8_t n)
Explanation	Delay for n×1ms.
Arguments	n: n×1ms
Return value	None
Remarks	None

[Function Name] mtr_start_motor

Synopsis	Start motor with 500rpm
Header	pi.h, main.h
Declaration	void mtr_start_motor(void)
Explanation	Start 4 channels of TAU0, obtain the rotation direction and start motor with 500rpm when Start/(Stop) key is pressed.
Arguments	None
Return value	None
Remarks	None

[Function Name] mtr_stop_motor

Synopsis	Stop motor
Header	pi.h, main.h
Declaration	void mtr_stop_motor(void)
Explanation	Stop 4 channels of TAU0 when (Start)/Stop key is pressed, motor will stop working.
Arguments	None
Return value	None
Remarks	None

[Function Name] mtr_current_detect

Synopsis	Detection of motor current
Header	ad.h, main.h
Declaration	void mtr_current_detect(void)
Explanation	Calculate the current value (mA) after sampling by A/D converter.
Arguments	None
Return value	None
Remarks	None

[Function Name] mtr_over_current_interrupt

Synopsis	Interrupt process for motor forced cut-off
Header	intp.h
Declaration	void mtr_over_current_interrupt(void)
Explanation	RTO turn to forced cut-off state after INTP0 is triggered by overcurrent, motor stop running. The user can add other rountine as required.
Arguments	None
Return value	None
Remarks	None

[Function Name] mtr_hall_a_interrupt

Synopsis	External interrpt process INTP1
Header	intp.h
Declaration	void mtr_hall_a_interrupt(void)
Explanation	Change the conduction pattern (RTO output) and enable speed measurement flag after detecting a variation of hall sensor signal a (both edges)
Arguments	None
Return value	None
Remarks	None

[Function Name] mtr_hall_b_interrupt

Synopsis	External interrpt process INTP2
Header	intp.h
Declaration	void mtr_hall_b_interrupt(void)
Explanation	Change the conduction pattern (RTO output) and enable speed measurement flag after detecting a variation of hall sensor signal b (both edges)
Arguments	None
Return value	None
Remarks	None

[Function Name] mtr_hall_c_interrupt

Synopsis	External interrpt process INTP3
Header	intp.h
Declaration	void mtr_hall_c_interrupt(void)
Explanation	Change the conduction pattern (RTO output) and enable speed measurement flag after detecting a variation of hall sensor signal c (both edges)
Arguments	None
Return value	None
Remarks	None

[Function Name] mtr_tau03_interrupt

Synopsis	1ms timer interrpt process TAU03
Header	tau0.h
Declaration	void mtr_tau03_interrupt(void)
Explanation	1ms timer, enable speed PI control flag per 5ms for speed PI control.
Arguments	None
Return value	None
Remarks	None

[Function Name] mtr_speed_calc

Synopsis	Caculate motor rotation speed
Header	pi.h, intp.h
Declaration	void mtr_speed_calc(void)
Explanation	Caculate motor rotation speed in interrpt process (INTP1, INTP2 and INTP3) by the counter value difference of TAU02
Arguments	None
Return value	None
Remarks	None

[Function Name] mtr_pi_ctrl_speed

Synopsis	Speed PI control
Header	pi.h, main.h
Declaration	void mtr_pi_ctrl_speed(void)
Explanation	Speed IP control per 5ms to make motor rotation speed and target speed (VR1) the same.
Arguments	None
Return value	None
Remarks	None

[Function Name] mtr_get_adc

Synopsis	A/D conversion
Header	ad.h, pi.h
Declaration	uint16_t mtr_get_adc(uint8_t ad_ch)
Explanation	Conversion channel selection for ANI3 (the sense voltage) or ANI7 (VR1 voltage).
Arguments	ad_ch: A/D conversion channel (3 or 7)
Return value	ad_temp: A/D conversion result
Remarks	None

[Function Name] mtr_eliminate_buffeting

Synopsis	Eliminate key buffeting
Header	tau0.h, main.h
Declaration	void mtr_eliminate_buffeting (void)
Explanation	Eliminate buffeting for Start/Stop key.
Arguments	None
Return value	button_flag: 0 (exists buffeting) or 1 (valid key)
Remarks	None

5.7 Flowcharts

5.7.1 Initialization Function

Figure 5.1 shows the flowchart for the initialization function.

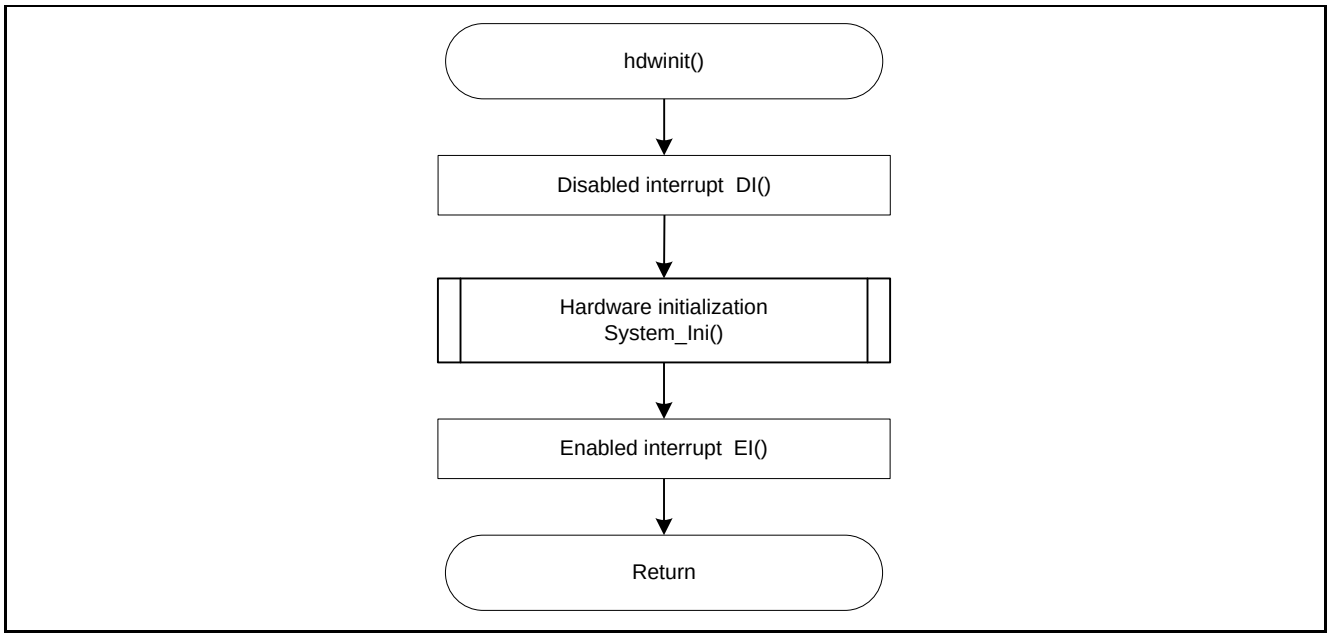

Figure 5.1 Initialization Function

5.7.2 System Function

Figure 5.2 shows the flowchart for the system function.

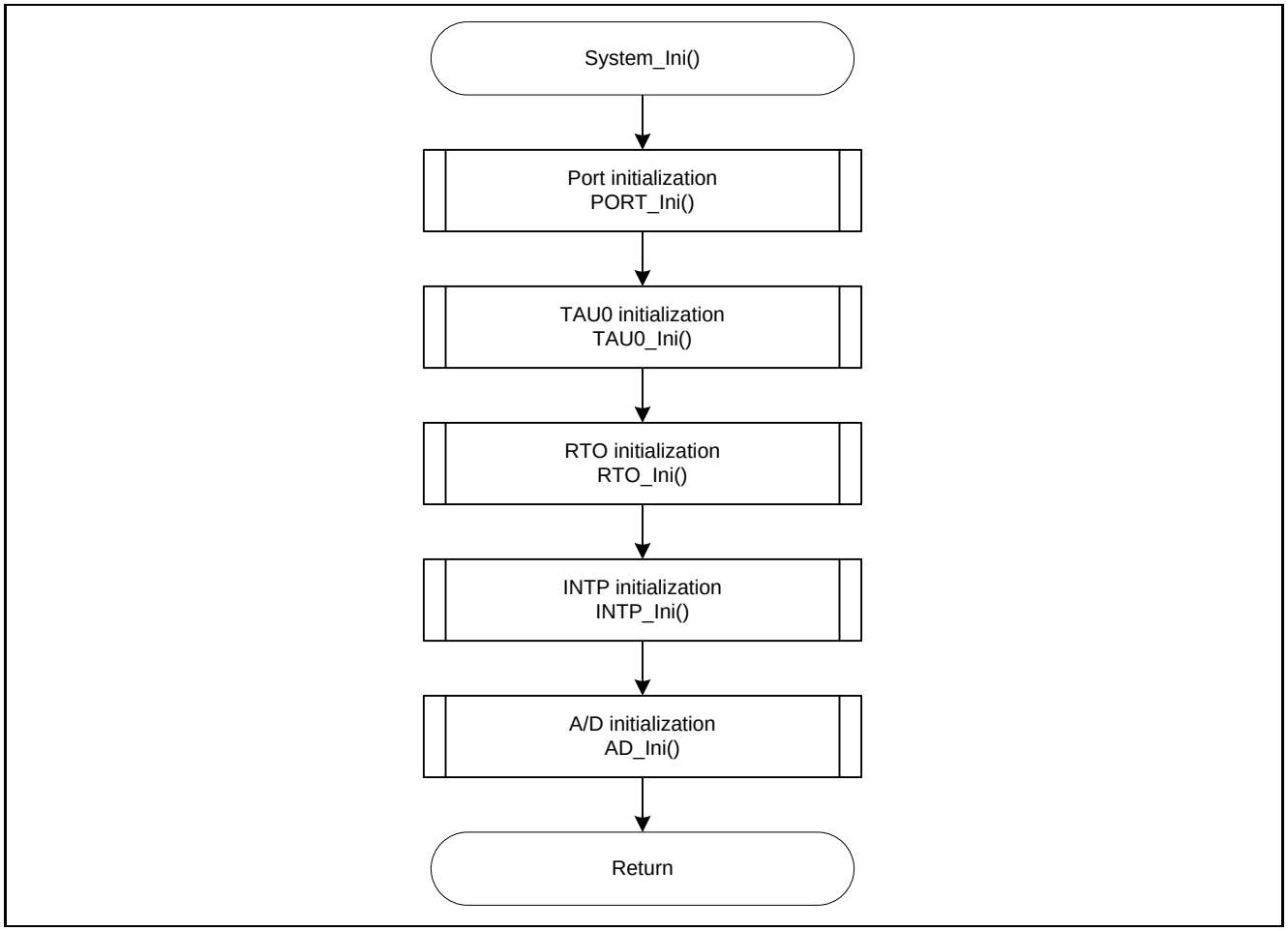

Figure 5.2 System Function

5.7.3 Initial Setting of I/O ports

Figure 5.3 shows the flowchart for initial setting of I/O ports.

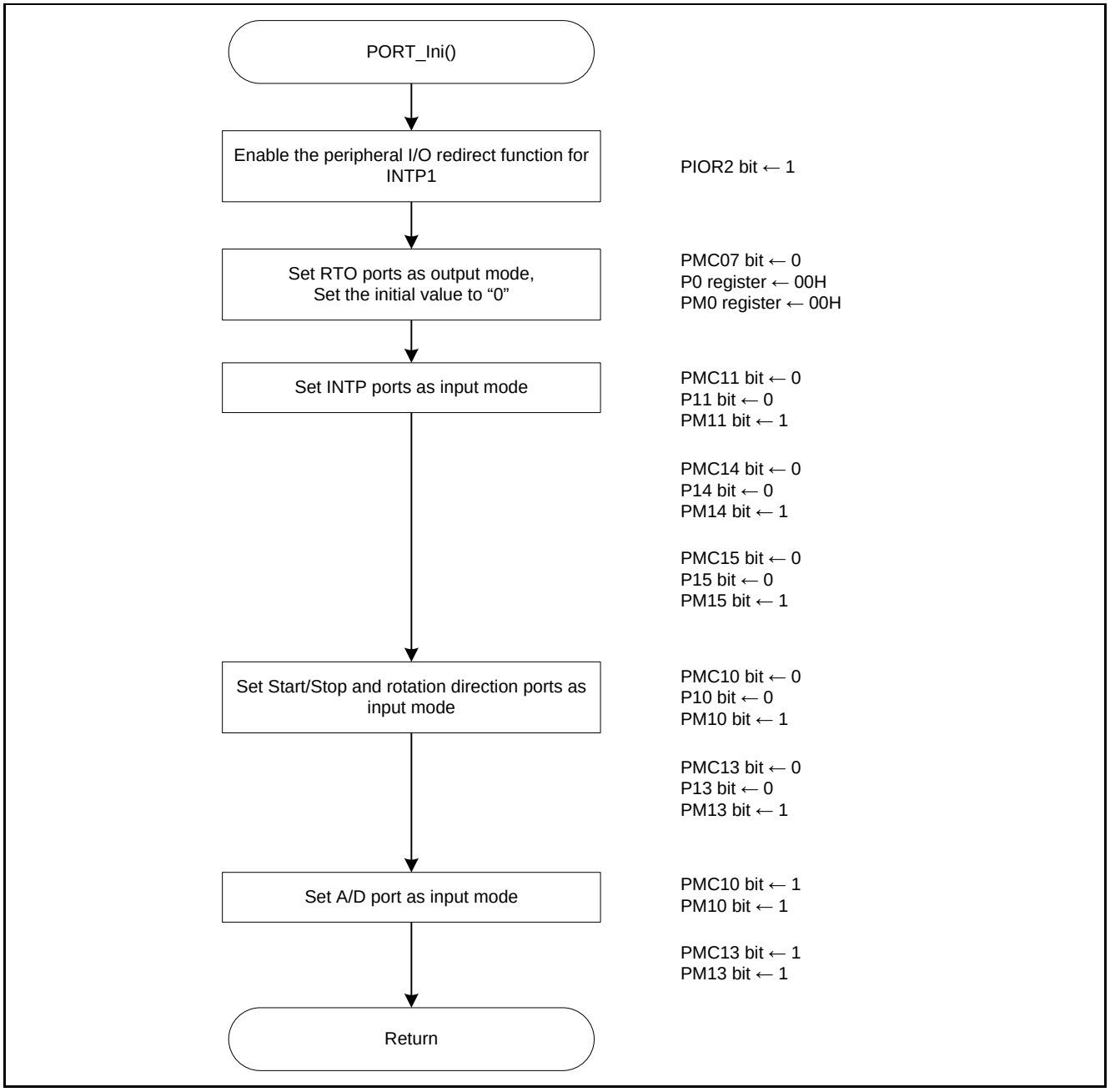

Figure 5.3 Initial Setting of I/O ports

Notes: Provide proper treatment for unused pins so that their electrical specifications are observed. Connect each of any unused input-only ports to V_{DD} or V_{SS} via separate resistors.

Set the port registers

- Peripheral I/O redirection register (PIOR)
  Set port as INTP1 input port.

Symbol: PIOR

7	6	5	4	3	2	1	0
PIOR7	0	0	0	0	PIOR2	PIOR1	PIOR0
x	-	-	-	-	**1**	x	x

Bit 2

PIOR2	Enable or disable external interrupt INTP1 redirects function
0	Disable redirects function INPT1: P00
1	**Enable redirects function** **INPT1: P03**

- Port mode control register 1 (PMC1)
  Set ports as digital I/O or analog input ports.

Symbol: PMC1

7	6	5	4	3	2	1	0
1	PMC16	PMC15	PMC14	PMC13	PMC12	PMC11	PMC10
-	**1**	**0**	**0**	**0**	**1**	**0**	**0**

Bit 6 and bit 2

PMC1n	P07 pin digital I/O/analog input selection (n = 2, 6)
0	Digital I/O (alternate function other than analog input)
1	**Analog input**

Bits 5 to 3 and bits 1 to 0

PMC1n	P1n pin digital I/O/analog input selection (n = 0, 1, 3 to 5)
0	**Digital I/O (alternate function other than analog input)**
1	Analog input

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

- Port register 0 (P0)
  Set the output latch value of a port.
- Port register 1 (P1)
  Set the output latch value of a port.

Symbol: P0

7	6	5	4	3	2	1	0
P07	P06	P05	P04	P03	P02	P01	P00
x	x	**0**	**0**	**0**	**0**	**0**	**0**

Bits 5 to 0

P0n	Output data control (in output mode) (n = 0 to 5)	Input data read (in input mode)
0	**Output 0**	Input low level
1	Output 1	Input high level

Symbol: P1

7	6	5	4	3	2	1	0
0	P16	P15	P14	P13	P12	P11	P10
-	**0**	**0**	**0**	**0**	**0**	**0**	**0**

Bits 6 to 0

P1n	Output data control (in output mode) (n = 0 to 6)	Input data read (in input mode)
0	**Output 0**	Input low level
1	Output 1	Input high level

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

- Port mode register 0 (PM0)
  Set output mode for the ports.
- Port mode register 1 (PM1)
  Set input mode for the ports.

Symbol: PM0

7	6	5	4	3	2	1	0
PM07	PM06	PM05	PM04	PM03	PM02	PM01	PM00
x	x	**0**	**0**	**0**	**0**	**0**	**0**

Bits 5 to 0

PM0n	PM0n pin I/O mode selection (n = 0 to 5)
0	**Output mode (output buffer on)**
1	Input mode (output buffer off)

Symbol: PM1

7	6	5	4	3	2	1	0
1	PM16	PM15	PM14	PM13	PM12	PM11	PM10
-	**1**	**1**	**1**	**1**	**1**	**1**	**1**

Bits 6 to 0

PM1n	PM1n pin I/O mode selection (n = 0 to 6)
0	Output mode (output buffer on)
1	**Input mode (output buffer off)**

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

5.7.4 Initial Setting of TAU

Figure 5.4 shows the flowchart for initial setting of TAU.

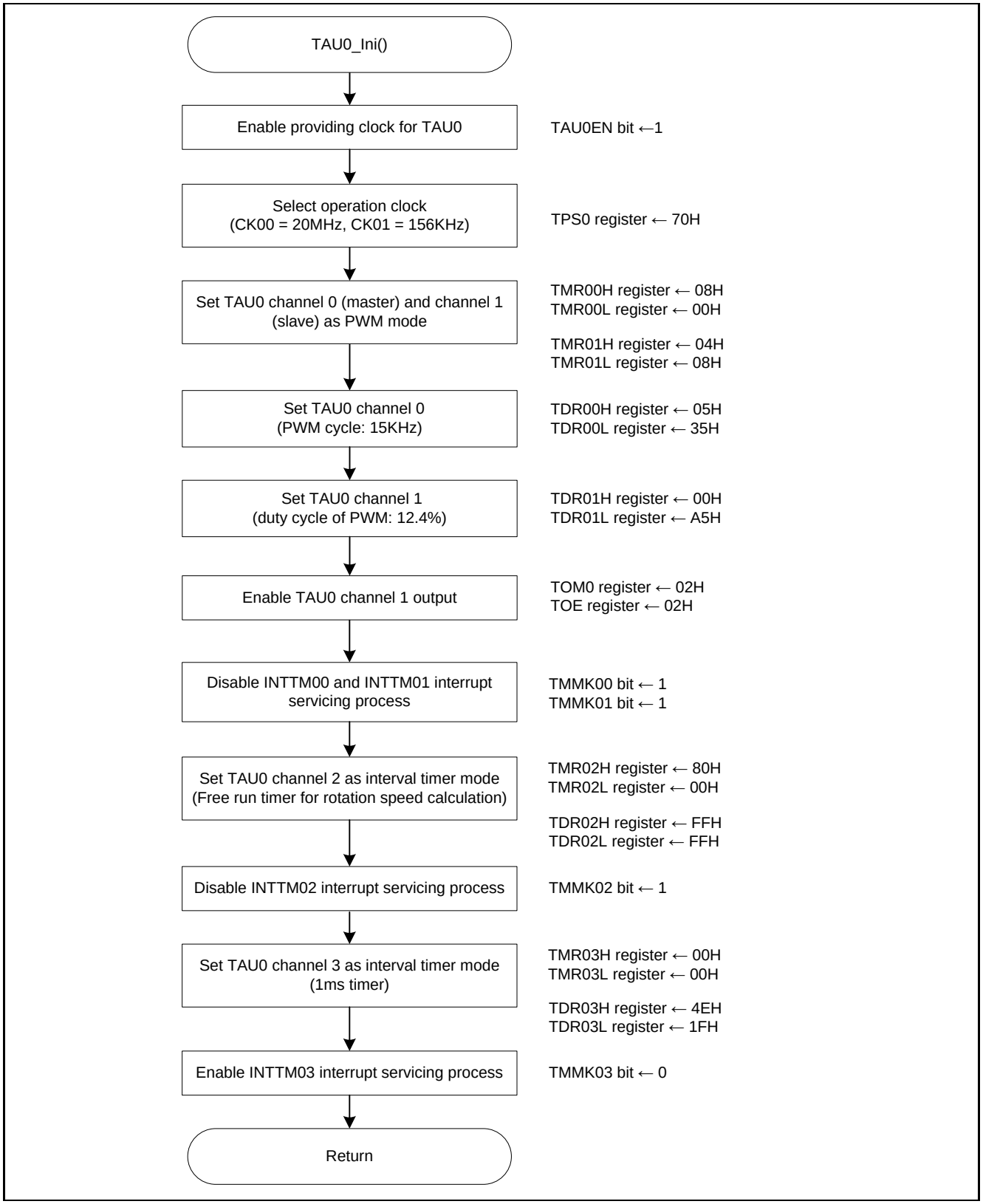

Figure 5.4 Initial Setting of TAU

Enable input clock supply for TAU0.

- Peripheral enable register 0 (PER0)
  Enable input clock supply for timer array unit 0.

Symbol: PER0

7	6	5	4	3	2	1	0
TMKAEN	RTOEN	ADCEN	0	0	SAU0EN	0	TAU0EN
x			-	-	x	-	**1**

Bit 0

TAU0EN	Control of TAU0 input clock supply
0	Stops input clock supply.
1	**Enables input clock supply.**

Select operation clock for TAU0.

- Timer clock select register 0 (TPS0)
  Select operation clock.

Symbol: TPS0

7	6	5	4	3	2	1	0
PRS013	PRS012	PRS011	PRS010	PRS003	PRS002	PRS001	PRS000
0	**1**	**1**	**1**	**0**	**0**	**0**	**0**

Bits 7 to 0

PRS 0k3	PRS 0k2	PRS 0k1	PRS 0k0		Selection of operation clock (CK0k) (k = 0, 1)				
					f_{CLK} = 1.25 MHz	f_{CLK} = 2.5 MHz	f_{CLK} = 5 MHz	f_{CLK} = 10 MHz	f_{CLK} = 20 MHz
0	**0**	**0**	**0**	**f_{CLK}**	1.25 MHz	2.5 MHz	5 MHz	10 MHz	**20 MHz**
0	0	0	1	$f_{CLK}/2$	625 kHz	1.25 MHz	2.5 MHz	5 MHz	10 MHz
0	0	1	0	$f_{CLK}/2^2$	313 kHz	625 kHz	1.25 MHz	2.5 MHz	5 MHz
0	0	1	1	$f_{CLK}/2^3$	156 kHz	313 kHz	625 kHz	1.25 MHz	2.5 MHz
0	1	0	0	$f_{CLK}/2^4$	78.1 kHz	156 kHz	313 kHz	625 kHz	1.25 MHz
0	1	0	1	$f_{CLK}/2^5$	39.1 kHz	78.1 kHz	156 kHz	313 kHz	625 kHz
0	1	1	0	$f_{CLK}/2^6$	19.5 kHz	39.1 kHz	78.1 kHz	156 kHz	313 kHz
0	**1**	**1**	**1**	**$f_{CLK}/2^7$**	9.77 kHz	19.5 kHz	39.1 kHz	78.1 kHz	**156 kHz**
1	0	0	0	$f_{CLK}/2^8$	4.88 kHz	9.77 kHz	19.5 kHz	39.1 kHz	78.1 kHz
1	0	0	1	$f_{CLK}/2^9$	2.44 kHz	4.88 kHz	9.77 kHz	19.5 kHz	39.1 kHz
1	0	1	0	$f_{CLK}/2^{10}$	1.22 kHz	2.44 kHz	4.88 kHz	9.77 kHz	19.5 kHz
1	0	1	1	$f_{CLK}/2^{11}$	610 Hz	1.22 kHz	2.44 kHz	4.88 kHz	9.77 kHz
1	1	0	0	$f_{CLK}/2^{12}$	305 Hz	610 Hz	1.22 kHz	2.44 kHz	4.88 kHz
1	1	0	1	$f_{CLK}/2^{13}$	153 Hz	305 Hz	610 Hz	1.22 kHz	2.44 kHz
1	1	1	0	$f_{CLK}/2^{14}$	76.3 Hz	153 Hz	305 Hz	610 Hz	1.22 kHz
1	1	1	1	$f_{CLK}/2^{15}$	38.1 Hz	76.3 Hz	153 Hz	305 Hz	610 Hz

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Set operation mode for TAU0 channel 0.

- Timer mode register 00 (TMR00H, TMR00L)
  Select operation clock (f_{MCK}).
  Select count clock.
  Set start trigger by software trigger.
  Set operation mode.

Symbol: TMR00H

7	6	5	4	3	2	1	0
CKS001	0	0	CCS00	0	STS002	STS001	STS000
0	-	-	**0**	-	**0**	**0**	**0**

Bit 7

CKS001	Selection of operation clock (f_{MCK}) of channel 0
0	**Operation clock CK00 set by timer clock select register 0 (TPS0)**
1	Operation clock CK01 set by timer clock select register 0 (TPS0)

Bit 4

CCS00	Selection of count clock (f_{TCLK}) of channel 0
0	**Operation clock (f_{MCK}) specified by the CKS001 bit**
1	Valid edge of input signal input from the T00 pin

Bits 2 to 0

STS 002	STS 001	STS 000	Setting of start trigger or capture trigger of channel 0
0	**0**	**0**	**Only software trigger start is valid (other trigger sources are unselected).**
0	0	1	Valid edge of the TI00 pin input is used as the start trigger and capture trigger.
0	1	0	Both the edges of the TI00 pin input are used as a start trigger and a capture trigger.
1	0	0	When the channel is used as a slave channel with the one-shot pulse output, PWM output function, or multiple PWM output function: The interrupt request signal of the master channel (INTTM00) is used as the start trigger.
1	1	0	When the channel is used as a slave channel in two-channel input with one-shot pulse output function: The interrupt request signal of the master channel (INTTM00) is used as the start trigger.
Other than above			Setting prohibited

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Symbol: TMR00L

7	6	5	4	3	2	1	0
CIS001	CIS000	0	0	MD003	MD002	MD001	MD000
x	x	-	-	**0**	**0**	**0**	**0**

Bits 3 to 1

MD 003	MD 002	MD 001	Setting of operation mode of channel 0	Corresponding function	Count operation of TCR
0	**0**	**0**	**Interval timer mode**	**Interval timer/Square wave output/Divider function/PWM output (master)**	**Down count**
0	1	0	Capture mode	Input pulse interval measurement/Two-channel input with one-shot pulse output function (slave)	Up count
0	1	1	Event counter mode	External event counter	Down count
1	0	0	One-count mode	Delay counter/One-shot pulse output/Two-channel input with one-shot pulse output function (master)/PWM output (slave)	Down count
1	1	0	Capture & one-count mode	Measurement of high-/low-level width of input signal	Up count
Other than above			Setting prohibited		
The operation of each mode changes depending on the operation of MD000 bit (see the table below).					

Operation mode (Value set by the MD003 to MD001 bits)	MD 000	Setting of starting counting and interrupt
• **Interval timer mode (0, 0, 0)** • Capture mode (0, 1, 0)	**0**	**Timer interrupt is not generated when counting is started (timer output does not change, either).**
	1	Timer interrupt is generated when counting is started (timer output also changes).
• Event counter mode (0, 1, 1)	0	Timer interrupt is not generated when counting is started (timer output does not change, either).
• One-count mode (1, 0, 0)	0	Start trigger is invalid during counting operation. At that time, a timer interrupt is not generated.
	1	Start trigger is valid during counting operation. At that time, a timer interrupt is not generated.
• Capture & one-count mode • (1, 1, 0)	0	Timer interrupt is not generated when counting is started (timer output does not change, either). Start trigger is invalid during counting operation. At that time, a timer interrupt is not generated.
Other than above		Setting prohibited

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Set operation mode for TAU0 channel 1.

- Timer mode register 01 (TMR01H, TMR01L)
  Select operation clock (f_{MCK}).
  Select count clock.
  Set start trigger by INTTM01.
  Set operation mode.

Symbol: TMR01H

7	6	5	4	3	2	1	0
CKS011	0	0	CCS01	SPLIT01	STS012	STS011	STS010
0	-	-	**0**	**0**	**1**	**0**	**0**

Bit 7

CKS011	Selection of operation clock (f_{MCK}) of channel 1
0	**Operation clock CK00 set by timer clock select register 0 (TPS0)**
1	Operation clock CK01 set by timer clock select register 0 (TPS0)

Bit 4

CCS01	Selection of count clock (f_{TCLK}) of channel 1
0	**Operation clock (f_{MCK}) specified by the CKS011 bit**
1	Valid edge of input signal input from the T01 pin

Bit 3

SPLIT01	Selection of 8 or 16-bit timer operation for channels 1
0	**Operates as 16-bit timer.**
1	Operates as 8-bit timer.

Bits 2 to 0

STS 012	STS 011	STS 010	Setting of start trigger or capture trigger of channel 1
0	0	0	Only software trigger start is valid (other trigger sources are unselected).
0	0	1	Valid edge of the TI01 pin input is used as the start trigger and capture trigger.
0	1	0	Both the edges of the TI01 pin input are used as a start trigger and a capture trigger.
1	**0**	**0**	**When the channel is used as a slave channel with the one-shot pulse output, PWM output function, or multiple PWM output function:** **The interrupt request signal of the master channel (INTTM00) is used as the start trigger.**
1	1	0	When the channel is used as a slave channel in two-channel input with one-shot pulse output function: The interrupt request signal of the master channel (INTTM00) is used as the start trigger.
Other than above			Setting prohibited

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Symbol: TMR01L

7	6	5	4	3	2	1	0
CIS011	CIS010	0	0	MD013	MD012	MD011	MD010
x	x	-	-	**1**	**0**	**0**	**0**

Bits 3 to 1

MD 013	MD 012	MD 011	Setting of operation mode of channel 1	Corresponding function	Count operation of TCR
0	0	0	Interval timer mode	Interval timer/Square wave output/Divider function/PWM output (master)	Down count
0	1	0	Capture mode	Input pulse interval measurement/Two-channel input with one-shot pulse output function (slave)	Up count
0	1	1	Event counter mode	External event counter	Down count
1	**0**	**0**	**One-count mode**	**Delay counter/One-shot pulse output/Two-channel input with one-shot pulse output function (master)/PWM output (slave)**	**Down count**
1	1	0	Capture & one-count mode	Measurement of high-/low-level width of input signal	Up count
Other than above			Setting prohibited		
The operation of each mode changes depending on the operation of MD010 bit (see the table below).					

Operation mode (Value set by the MD013 to MD011 bits)	MD 010	Setting of starting counting and interrupt
• Interval timer mode (0, 0, 0) • Capture mode (0, 1, 0)	0	Timer interrupt is not generated when counting is started (timer output does not change, either).
	1	Timer interrupt is generated when counting is started (timer output also changes).
• Event counter mode (0, 1, 1)	0	Timer interrupt is not generated when counting is started (timer output does not change, either).
• **One-count mode (1, 0, 0)**	**0**	**Start trigger is invalid during counting operation. At that time, a timer interrupt is not generated.**
	1	Start trigger is valid during counting operation. At that time, a timer interrupt is not generated.
• Capture & one-count mode • (1, 1, 0)	0	Timer interrupt is not generated when counting is started (timer output does not change, either). Start trigger is invalid during counting operation. At that time, a timer interrupt is not generated.
Other than above		Setting prohibited

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Set PWM period.

- Timer data register 00 (TDR00H, TDR00L)
  Set interval counter of PWM period.

Symbol: TDR00H

7	6	5	4	3	2	1	0
0	0	0	0	0	1	0	1

Symbol: TDR00L

7	6	5	4	3	2	1	0
0	0	1	1	0	1	0	1

Pulse period = {Set value of TDR00 (master) + 1} × Count clock period

= (0x0535+1) × 1/20MHz = 66.7us (15KHz)

Set PWM duty cycle.

- Timer data register 01 (TDR01H, TDR01L)
  Set interval counter of PWM duty cycle.

Symbol: TDR01H

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0

Symbol: TDR01L

7	6	5	4	3	2	1	0
1	0	1	0	0	1	0	1

Duty cycle [%] = {Set value of TDR01 (slave)}/{Set value of TDR00 (master) + 1} × 100

= 0x00a5/(0x0535+1) × 100 = 12.4%

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Select channel 1 output.

- Timer output mode register 0 (TOM0)
  Timer output mode control.

Symbol: TOM0

7	6	5	4	3	2	1	0
0	0	0	0	TOM03	TOM02	TOM01	0
-	-	-	-	x	x	**1**	-

Bit 1

TOM01	Control of timer output mode of channel 1
0	Used as the independent channel operation function (to produce toggle output by the interrupt request signal (INTTM01))
1	**Slave channel output mode (output is set by the interrupt request signal (INTTM00) of the master channel, and reset by the timer interruptrequest signal (INTTM0) of the slave channel)**

Enable timer output

- Timer output enable register 0 (TOE0)
  Enable or disable timer output of each channel.

Symbol: TOE0

7	6	5	4	3	2	1	0
0	0	0	0	TOE03	TOE02	TOE01	TOE00
-	-	-	-	x	x	**1**	x

Bit 1

TOE01	Timer output enable/disable of channel 1
0	Disable output of timer. Without reflecting on TO01 bit timer operation, to fixed the output. Writing to the TO01 bit is enabled and the level set inthe TO01 bit is output from the TO01 pin.
1	**Enable output of timer.** **Reflected in the TO0 bit timer operation, to generate the output waveform.** **Writing to the TO0 bit is disabled (writing is ignored).**

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Disable interrupt.

- Interrupt mask flag registers (MK0L, MK0H)
  Interrupt servicing control.

Symbol: MK0L

7	6	5	4	3	2	1	0
TMMK00	TMMK01H	SREMK0	SRMK0	STMK0 CSIMK00	PMK1	PMK0	WDTIMK
1	x	x	x	x			x

Bit 7

TMMK00	Interrupt servicing control
0	Interrupt servicing enabled
1	**Interrupt servicing disabled**

Symbol: MK0H

7	6	5	4	3	2	1	0
TMMK02	1	TMMK03H	PMK3	PMK2	KRMK	ADMK	TMMK01
	-	x			x		**1**

Bit 0

TMMK01	Interrupt servicing control
0	Interrupt servicing enabled
1	**Interrupt servicing disabled**

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Set operation mode for TAU0 channel 2.

- Timer mode register 02 (TMR02H, TMR02L)
  Select operation clock (f_{MCK}).
  Select count clock.
  Set start trigger as software trigger.
  Set operation mode.

Symbol: TMR02H

7	6	5	4	3	2	1	0
CKS021	0	0	CCS02	MASTER02	STS022	STS021	STS020
1	-	-	**0**	**0**	**0**	**0**	**0**

Bit 7

CKS021	Selection of operation clock (f_{MCK}) of channel 2
0	Operation clock CK00 set by timer clock select register 0 (TPS0)
1	**Operation clock CK01 set by timer clock select register 0 (TPS0)**

Bit 4

CCS02	Selection of count clock (f_{TCLK}) of channel 2
0	**Operation clock (f_{MCK}) specified by the CKS021 bit**
1	Valid edge of input signal input from the T02 pin

Bit 3

MASTER02	Selection of independent channel operation/simultaneous channel operation (slave/master) of channel 2
0	**Operates as the slave channel in the independent channel operation function or the simultaneous channel operation function.**
1	Operates as the master channel in the simultaneous channel operation function.

Bits 2 to 0

STS 022	STS 021	STS 020	Setting of start trigger or capture trigger of channel 2
0	**0**	**0**	**Only software trigger start is valid (other trigger sources are unselected).**
0	0	1	Valid edge of the TI02 pin input is used as the start trigger and capture trigger.
0	1	0	Both the edges of the TI02 pin input are used as a start trigger and a capture trigger.
1	0	0	When the channel is used as a slave channel with the one-shot pulse output, PWM output function, or multiple PWM output function: The interrupt request signal of the master channel (INTTM00) is used as the start trigger.
1	1	0	When the channel is used as a slave channel in two-channel input with one-shot pulse output function: The interrupt request signal of the master channel (INTTM00) is used as the start trigger.
Other than above			Setting prohibited

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Symbol: TMR02L

7	6	5	4	3	2	1	0
CIS021	CIS020	0	0	MD023	MD022	MD021	MD020
x	x	-	-	0	0	0	0

Bits 3 to 1

MD 023	MD 022	MD 021	Setting of operation mode of channel 2	Corresponding function	Count operation of TCR
0	**0**	**0**	**Interval timer mode**	**Interval timer/Square wave output/Divider function/PWM output (master)**	**Down count**
0	1	0	Capture mode	Input pulse interval measurement/Two-channel input with one-shot pulse output function (slave)	Up count
0	1	1	Event counter mode	External event counter	Down count
1	0	0	One-count mode	Delay counter/One-shot pulse output/Two-channel input with one-shot pulse output function (master)/PWM output (slave)	Down count
1	1	0	Capture & one-count mode	Measurement of high-/low-level width of input signal	Up count
Other than above			Setting prohibited		
The operation of each mode changes depending on the operation of MD020 bit (see the table below).					

Operation mode (Value set by the MD023 to MD021 bits)	MD 020	Setting of starting counting and interrupt
• **Interval timer mode (0, 0, 0)**	**0**	**Timer interrupt is not generated when counting is started (timer output does not change, either).**
• Capture mode (0, 1, 0)	1	Timer interrupt is generated when counting is started (timer output also changes).
• Event counter mode (0, 1, 1)	0	Timer interrupt is not generated when counting is started (timer output does not change, either).
• One-count mode (1, 0, 0)	0	Start trigger is invalid during counting operation. At that time, a timer interrupt is not generated.
	1	Start trigger is valid during counting operation. At that time, a timer interrupt is not generated.
• Capture & one-count mode • (1, 1, 0)	0	Timer interrupt is not generated when counting is started (timer output does not change, either). Start trigger is invalid during counting operation. At that time, a timer interrupt is not generated.
Other than above		Setting prohibited

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Set free run timer.

- Timer data register 02(TDR02H, TDR02L)
  Set free run timer period.

Symbol: TDR02H

7	6	5	4	3	2	1	0
1	**1**	**1**	**1**	**1**	**1**	**1**	**1**

Symbol: TDR02L

7	6	5	4	3	2	1	0
1	**1**	**1**	**1**	**1**	**1**	**1**	**1**

Timer period = {Set value of TDR02 + 1} × Count clock period

= (0xffff+1) × 1/156KHz = 420ms

Disable interrupt.

- Interrupt mask flag registers (MK0H)
  Interrupt servicing control.

Symbol: MK0H

7	6	5	4	3	2	1	0
TMMK02	1	TMMK03H	PMK3	PMK2	KRMK	ADMK	TMMK01
1	-	x			x		

Bit 7

TMMK02	Interrupt servicing control
0	Interrupt servicing enabled
1	**Interrupt servicing disabled**

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Set operation mode for TAU0 channel 3.

- Timer mode register 03 (TMR03H, TMR03L)
  Select operation clock (f_{MCK}).
  Select count clock.
  Set start trigger as software trigger.
  Set operation mode.

Symbol: TMR03H

7	6	5	4	3	2	1	0
CKS031	0	0	CCS03	SPLIT03	STS032	STS031	STS030
0	-	-	**0**	**0**	**0**	**0**	**0**

Bit 7

CKS031	Selection of operation clock (f_{MCK}) of channel 3
0	**Operation clock CK00 set by timer clock select register 0 (TPS0)**
1	Operation clock CK01 set by timer clock select register 0 (TPS0)

Bit 4

CCS03	Selection of count clock (f_{TCLK}) of channel 3
0	**Operation clock (f_{MCK}) specified by the CKS031 bit**
1	Valid edge of input signal input from the T013 pin

Bit 3

SPLIT03	Selection of 8 or 16-bit timer operation for channels 3
0	**Operates as 16-bit timer.**
1	Operates as 8-bit timer.

Bits 2 to 0

STS 032	STS 031	STS 030	Setting of start trigger or capture trigger of channel 3
0	**0**	**0**	**Only software trigger start is valid (other trigger sources are unselected).**
0	0	1	Valid edge of the TI03 pin input is used as the start trigger and capture trigger.
0	1	0	Both the edges of the TI03 pin input are used as a start trigger and a capture trigger.
1	0	0	When the channel is used as a slave channel with the one-shot pulse output, PWM output function, or multiple PWM output function: The interrupt request signal of the master channel (INTTM00) is used as the start trigger.
1	1	0	When the channel is used as a slave channel in two-channel input with one-shot pulse output function: The interrupt request signal of the master channel (INTTM00) is used as the start trigger.
Other than above			Setting prohibited

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Symbol: TMR03L

7	6	5	4	3	2	1	0
CIS031	CIS030	0	0	MD033	MD032	MD031	MD030
x	x	-	-	**0**	**0**	**0**	**0**

Bits 3 to 1

MD 033	MD 032	MD 031	Setting of operation mode of channel 3	Corresponding function	Count operation of TCR
0	**0**	**0**	**Interval timer mode**	**Interval timer/Square wave output/Divider function/PWM output (master)**	**Down count**
0	1	0	Capture mode	Input pulse interval measurement/Two-channel input with one-shot pulse output function (slave)	Up count
0	1	1	Event counter mode	External event counter	Down count
1	0	0	One-count mode	Delay counter/One-shot pulse output/Two-channel input with one-shot pulse output function (master)/PWM output (slave)	Down count
1	1	0	Capture & one-count mode	Measurement of high-/low-level width of input signal	Up count
Other than above			Setting prohibited		
The operation of each mode changes depending on the operation of MD030 bit (see the table below).					

Operation mode (Value set by the MD033 to MD031 bits)	MD 030	Setting of starting counting and interrupt
• **Interval timer mode (0, 0, 0)**	**0**	**Timer interrupt is not generated when counting is started (timer output does not change, either).**
• Capture mode (0, 1, 0)	1	Timer interrupt is generated when counting is started (timer output also changes).
• Event counter mode (0, 1, 1)	0	Timer interrupt is not generated when counting is started (timer output does not change, either).
• One-count mode (1, 0, 0)	0	Start trigger is invalid during counting operation. At that time, a timer interrupt is not generated.
	1	Start trigger is valid during counting operation. At that time, a timer interrupt is not generated.
• Capture & one-count mode • (1, 1, 0)	0	Timer interrupt is not generated when counting is started (timer output does not change, either). Start trigger is invalid during counting operation. At that time, a timer interrupt is not generated.
Other than above		Setting prohibited

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Set interval timer period (1ms).

- Timer data register 03 (TDR03H, TDR03L)
  Set interval counter of timer period.

Symbol: TDR03H

7	6	5	4	3	2	1	0
0	**1**	**0**	**0**	**1**	**1**	**1**	**0**

Symbol: TDR03L

7	6	5	4	3	2	1	0
0	**0**	**0**	**1**	**1**	**1**	**1**	**1**

Timer period = {Set value of TDR03 + 1} × Count clock period
= (0x4e1f+1) × 1/20MHz = 1ms

Set timer interrupt.

- Interrupt request flag registers (IF1L)
  Clear interrupt request flag.
- Interrupt mask flag registers (MK1L)
  Enable interrupt servicing.

Symbol: IF1L

7	6	5	4	3	2	1	0
0	0	0	0	PIF5	PIF4	ITIF	TMIF03
-	-	-	-	x	x	x	**0**

Bit 0

TMIF03	Interrupt request flag
0	**No interrupt request signal is generated**
1	Interrupt request is generated, interrupt request status

Symbol: MK1L

7	6	5	4	3	2	1	0
1	1	1	1	PMK5	PMK4	ITMK	TMMK03
-	-	-	-	x	x	x	**0**

Bit 0

TMMK03	Interrupt servicing control
0	**Interrupt servicing enabled**
1	Interrupt servicing disabled

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

5.7.5 Initial Setting of RTO

Figure 5.5 shows the flowchart for initial setting of RTO.

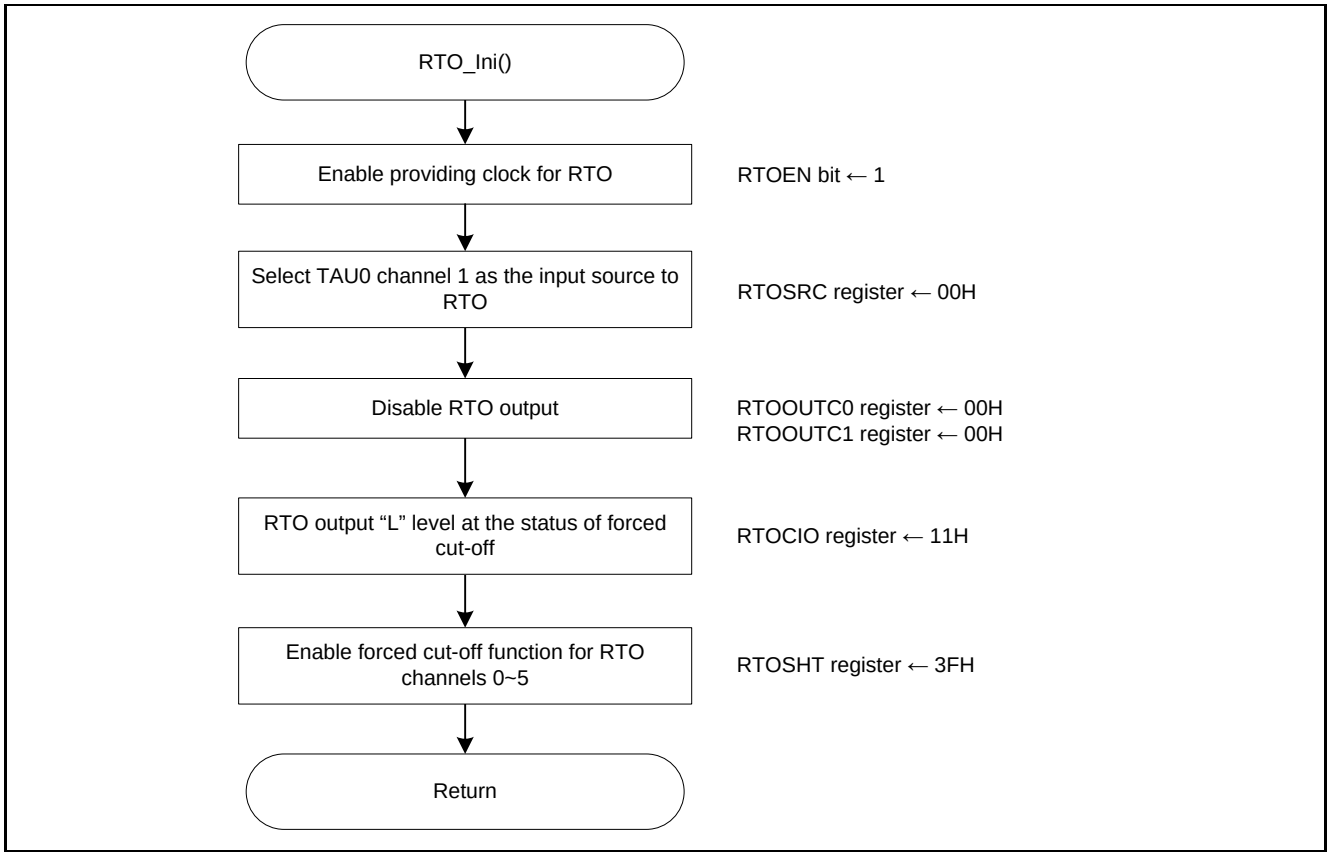

Figure 5.5 Initial Setting of RTO

Enable input clock supply for real-time output controller (RTO).

- Peripheral enable register 0 (PER0)
  Enable input clock supply for RTO.

Symbol: PER0

7	6	5	4	3	2	1	0
TMKAEN	RTOEN	ADCEN	0	0	SAU0EN	0	TAU0EN
x	**1**		-	-	x	-	

Bit 6

RTOEN	Control of RTO input clock supply
0	Stops input clock supply.
1	**Enables input clock supply.**

Select source for RTO.

- RTO source selection register (RTOSRC)
  RTO source selection control.

Symbol: RTOSRC

7	6	5	4	3	2	1	0
RTOSRC7	RTOSRC6	RTOSRC5	RTOSRC4	RTOSRC3	RTOSRC2	RTOSRC1	RTOSRC0
x	x	**0**	**0**	**0**	**0**	**0**	**0**

Bits 5 to 0

RTOSRCn	Selection of RTIO0n output source (n = 0 to 5)
0	**Select TO01.**
1	Select TO03.

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Set RTO output waveforms.

- RTO control register 0 (RTOOUTC0)
- RTO control register 1 (RTOOUTC1)
  Output inverting control.
  Enable RTO output.

Symbol: RTOOUTC0

7	6	5	4	3	2	1	0
RTOACT3	RTOACT2	RTOACT1	RTOACT0	RTOSEL3	RTOSEL2	RTOSEL1	RTOSEL0
0	**0**	**0**	**0**	**0**	**0**	**0**	**0**

Bits 7 to 4

RTOACTn	RTIO0n output inverting control (n = 0 to 3)
0	**Do not invert.**
1	Invert.

Bits 3 to 0

RTOSELn	RTIO0n output control (n = 0 to 3)
0	**Disable output.**
1	Enable output..

Symbol: RTOOUTC1

7	6	5	4	3	2	1	0
RTOACT7	RTOACT6	RTOACT5	RTOACT4	RTOSEL7	RTOSEL6	RTOSEL5	RTOSEL4
x	x	**0**	**0**	x	x	**0**	**0**

Bits 5 to 4

RTOACTn	RTIO0n output inverting control (n = 4 to 5)
0	**Do not invert.**
1	Invert.

Bits 1 to 0

RTOSELn	RTIO0n output control (n = 4 to 5)
0	**Disable output.**
1	Enable output..

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Output status of forced cutoff.

- RTO forced cutoff output selection register (RTOCIO)
  Forced cutoff output selection control.

Symbol: RTOCIO

7	6	5	4	3	2	1	0
RTOCIO7	RTOCIO6	RTOCIO5	RTOCIO4	RTOCIO3	RTOCIO2	RTOCIO1	RTOCIO0
x	x	**0**	**1**	x	x	**0**	**1**

Bits 5 to 4

RTOCIO5	RTOCIO4	Selection of the status of forced cutoff output from RTIO05 to RTIO03
0	0	Hi-Z output
0	**1**	**Low-level output**
1	0	High-level output
1	1	Cutoff-invalidated

Bits 1 to 0

RTOCIO1	RTOCIO0	Selection of the status of forced cutoff output from RTIO02 to RTIO00
0	0	Hi-Z output
0	**1**	**Low-level output**
1	0	High-level output
1	1	Cutoff-invalidated

Enable fourced cutoff output.

- RTO forced cutoff control register (RTOSHT)
  Forced cutoff enable control.

Symbol: RTOSHT

7	6	5	4	3	2	1	0
RTOSHT7	RTOSHT6	RTOSHT5	RTOSHT4	RTOSHT3	RTOSHT2	RTOSHT1	RTOSHT0
x	x	**1**	**1**	**1**	**1**	**1**	**1**

Bits 5 to 0

RTOSHTn	RTIO0n output forced cutoff control (n = 0 to 5)
0	Disable forced cutoff.
1	**Enable forced cutoff.**

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

5.7.6 Initial Setting of INTP

Figure 5.6 shows the flowchart for initial setting of INTP.

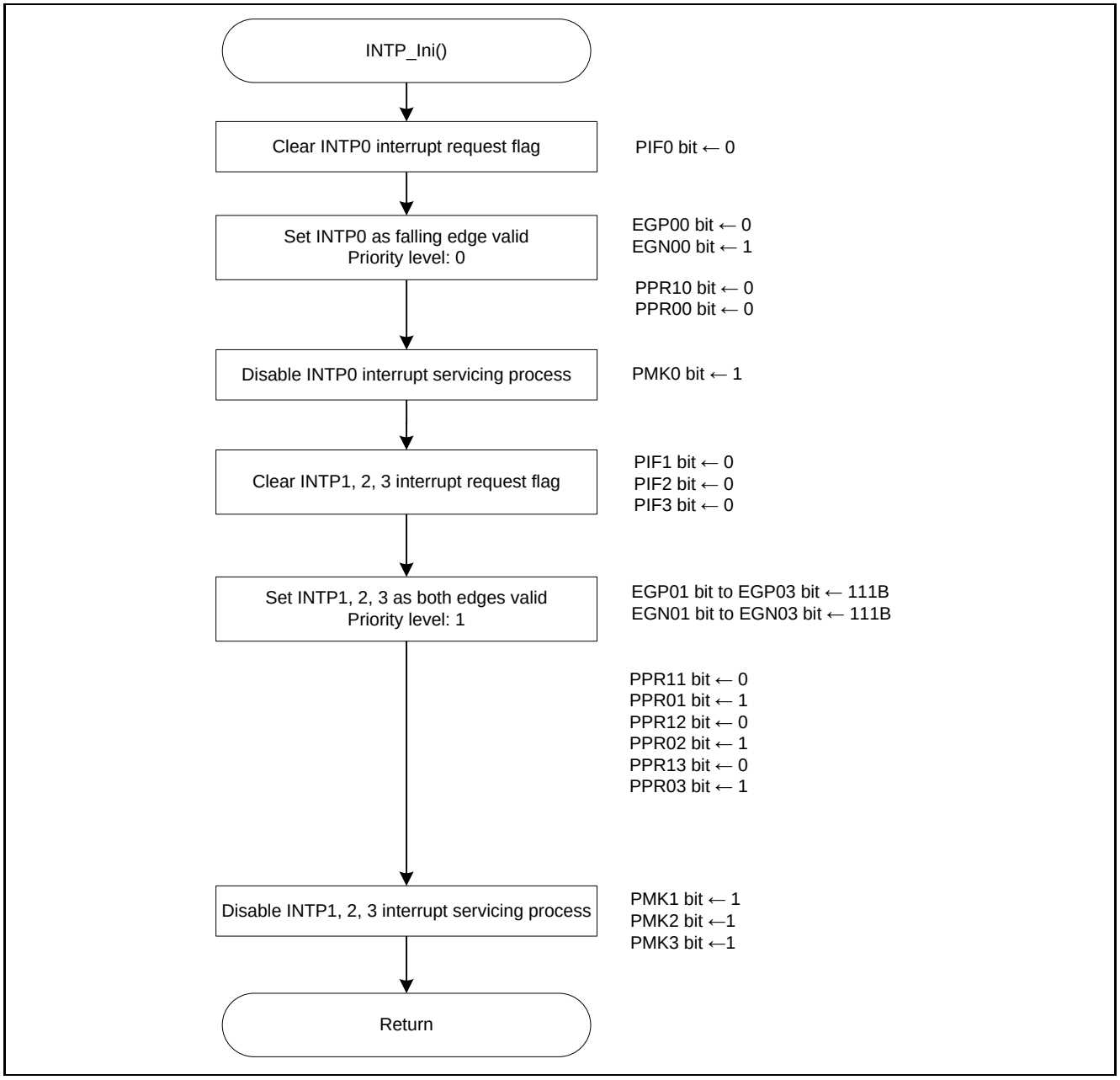

Figure 5.6 Initial Setting of INTP

Set external interrupt

- External interrupt rising edge enable register 0 (EGP0)
- External interrupt falling edge enable register 0 (EGN0)
  Specify the valid edge for INTP.
- Interrupt request flag register (IF0H, IF0L)
  Clear interrupt request flag.
- Interrupt mask flag register (MK0H, MK0L)
  Disable interrupt servicing.
- Priority specification flag registers (PR10L, PR00L, PR10H, PR00H)
  Set priority level.

Symbol: EGP0

7	6	5	4	3	2	1	0
0	0	EGP5	EGP4	EGP3	EGP2	EGP1	EGP0
-	-	x	x	**1**	**1**	**1**	**0**

Symbol: EGN0

7	6	5	4	3	2	1	0
0	0	EGN5	EGN4	EGN3	EGN2	EGN1	EGN0
-	-	x	x	**1**	**1**	**1**	**1**

Bits 3 to 1

EGPn	EGNn	INTP1, 2, 3 pins valid edge selection (n = 1 to 3)
0	0	Edge detection disabled.
0	1	Falling edge.
1	0	Rising edge.
1	**1**	**Both rising and falling edges.**

Bit 0

EGP0	EGN0	INTP0 pin valid edge selection
0	0	Edge detection disabled.
0	**1**	**Falling edge.**
1	0	Rising edge.
1	1	Both rising and falling edges.

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Symbol: IF0L

7	6	5	4	3	2	1	0
TMIF00	TMIF01H	SREIF0	SRIF0	STIF0 CSIIF00	PIF1	PIF0	WDTIF
	x	x	x	x	**0**	**0**	x

Bits 2 to 1

PIFn	Interrupt request flag (n = 0 to 1)
0	**No interrupt request signal is generated**
1	Interrupt request is generated, interrupt request status

Symbol: IF0L

7	6	5	4	3	2	1	0
TMIF02	0	TMIF03H	PIF3	PIF2	KRIF	ADIF	TMIF01
	-	x	**0**	**0**	x	x	

Bits 4 to 3

PIFn	Interrupt request flag (n = 2 to 3)
0	**No interrupt request signal is generated**
1	Interrupt request is generated, interrupt request status

Symbol: MK0L

7	6	5	4	3	2	1	0
TMMK00	TMMK01H	SREMK0	SRMK0	STMK0 CSIMK00	PMK1	PMK0	WDTIMK
	x	x	x	x	**1**	**1**	x

Bits 2 to 0

PMKn	Interrupt servicing control (n = 0 to 1)
0	Interrupt servicing enabled
1	**Interrupt servicing disabled**

Symbol: MK0H

7	6	5	4	3	2	1	0
TMMK02	1	TMMK03H	PMK3	PMK2	KRMK	ADMK	TMMK01
	-	x	**1**	**1**	x		

Bits 4 to 3

PMKn	Interrupt servicing control (n = 2 to 3)
0	Interrupt servicing enabled
1	**Interrupt servicing disabled**

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Symbol: PR00L

7	6	5	4	3	2	1	0
TMPR000	TMPR001H	SREPR00	SRPR00	STPR00 CSIPR000	PPR01	PPR00	WDTIPR0
x	x	x	x	x	**1**	**0**	x

Symbol: PR10L

7	6	5	4	3	2	1	0
TMPR100	TMPR101H	SREPR10	SRPR10	STPR10 CSIPR100	PPR11	PPR10	WDTIPR1
x	x	x	x	x	**0**	**0**	x

Bit 1

PPR10	PPR00	Priority level selection for INTP0
0	**0**	**Specifying level 0 (high priority)**
0	1	Specifying level 1
1	0	Specifying level 2
1	1	Specifying level 3 (low priority)

Bit 2

PPR11	PPR01	Priority level selection for INTP1
0	0	Specifying level 0 (high priority)
0	**1**	**Specifying level 1**
1	0	Specifying level 2
1	1	Specifying level 3 (low priority)

Symbol: PR00H

7	6	5	4	3	2	1	0
TMPR02	1	TMPR03H	PPR03	PPR02	KRPR0	ADPR0	TMPR001
x	-	x	**1**	**1**	x	x	x

Symbol: PR10H

7	6	5	4	3	2	1	0
TMPR102	1	TMPR103H	PPR13	PPR12	KRPR1	ADPR1	TMPR101
x	-	x	**0**	**0**	x	x	x

Bits 4 to 3

PPR1n	PPR0n	Priority level selection for INTP2 and INTP3 (n = 2 to 3)
0	0	Specifying level 0 (high priority)
0	**1**	**Specifying level 1**
1	0	Specifying level 2
1	1	Specifying level 3 (low priority)

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

5.7.7 Initial Setting of A/D Converter

Figure 5.7 shows the flowchart for initial setting of A/D converter.

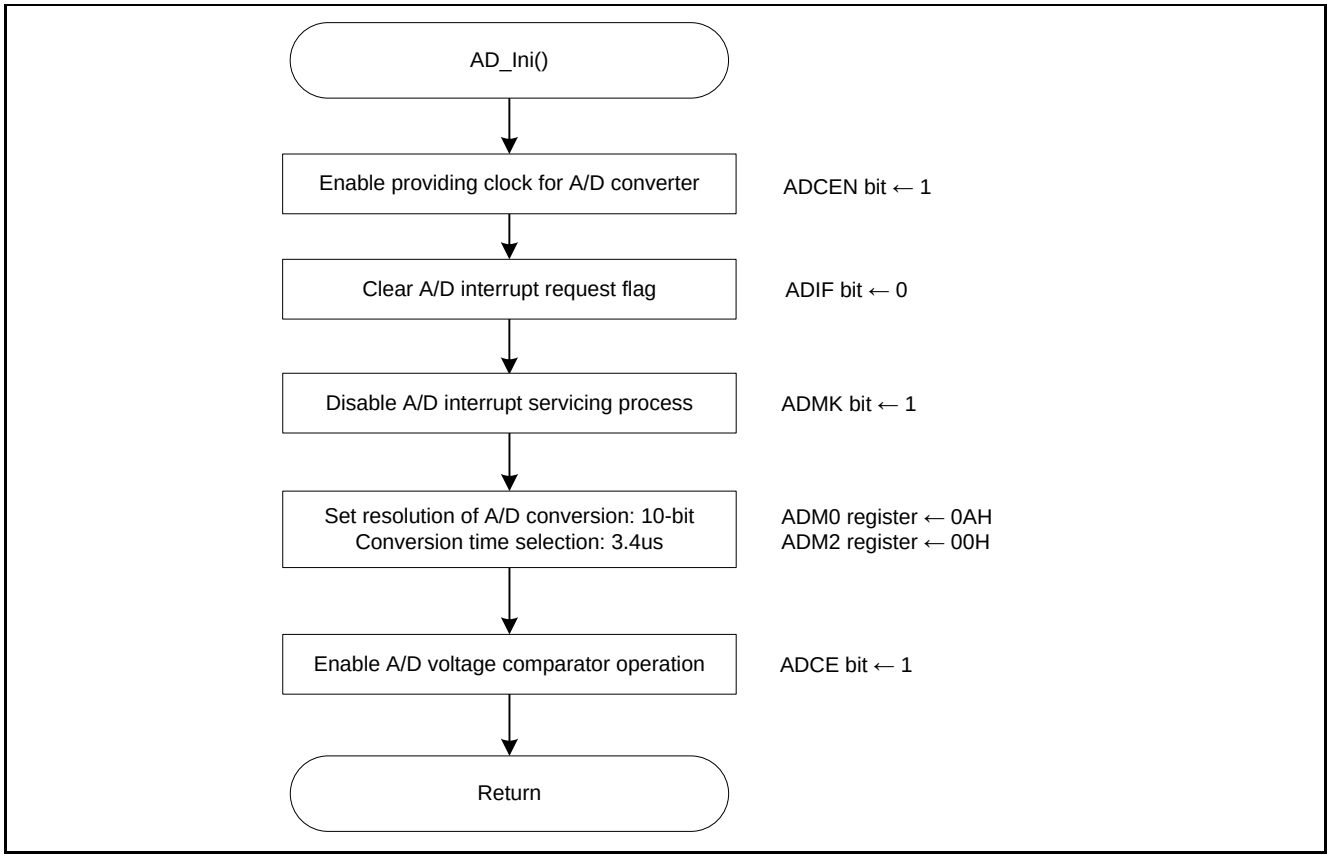

Figure 5.7 Initial Setting of A/D Converter

Set A/D converter

- Peripheral enable register 0 (PER0)
  Enable input clock supply for A/D converter.
- A/D converter mode register 0 (ADM0)
- A/D converter mode register 2 (ADM2)
  Specify time and resolution of A/D conversion.

Symbol: PER0

7	6	5	4	3	2	1	0
TMKAEN	RTOEN	ADCEN	0	0	SAU0EN	0	TAU0EN
x		**1**	-	-	x	-	

Bit 5

ADCEN	Control of A/D conversion input clock supply
0	Stops input clock supply.
1	**Enables input clock supply.**

Symbol: ADM0

7	6	5	4	3	2	1	0
ADCS	0	0	FR1	FR0	0	LV0	ADCE
0	-	-	**0**	**1**	-	**1**	**0**

Bit 7

ADCS	A/D conversion operation control
0	**Stops conversion operation (conversion stopped/standby status)**
1	Enables conversion operation (conversion operation status)

Bit 4, bit 3 and bit 1

ADM0			Conversion Clock	Number of Conversion Clock	Conversion Time	Conversion Time Selection (us)				
FR 1	FR 0	LV 0				f_{CLK}= 1.25MHz	f_{CLK}= 2.5MHz	f_{CLK}= 5MHz	f_{CLK}= 10MHz	f_{CLK}= 20MHz
0	0	0	$f_{CLK}/8$	23 f_{AD} (Number of sampling clock: 9 f_{AD})	$184/f_{CLK}$	Setting prohibited	Setting prohibited	Setting prohibited	18.4	9.2
0	1		$f_{CLK}/4$		$92/f_{CLK}$			18.4	9.2	4.6
1	0		$f_{CLK}/2$		$46/f_{CLK}$		18.4	9.2	4.6	Setting prohibited
1	1		f_{CLK}		$23/f_{CLK}$	18.4	9.2	4.6	Setting prohibited	
0	0	1	$f_{CLK}/8$	**17 f_{AD} (Number of sampling clock: 3 f_{AD})**	$136/f_{CLK}$	Setting prohibited	Setting prohibited	Setting prohibited	13.6	6.8
0	**1**		$f_{CLK}/4$		$68/f_{CLK}$			13.6	6.8	3.4
1	0		**$f_{CLK}/2$**		**$34/f_{CLK}$**		13.6	6.8	**3.4**	Setting prohibited
1	1		f_{CLK}		$17/f_{CLK}$	13.6	6.8	3.4	Setting prohibited	

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Symbol: ADM0

7	6	5	4	3	2	1	0
ADCS	0	0	FR1	FR0	0	LV0	ADCE
0	-	-	**0**	**1**	-	**1**	**0**

Bit 0

ADCE	A/D voltage comparator operation control
0	**Stops A/D voltage comparator operation**
1	Enables A/D voltage comparator operation

Symbol: ADM2

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	ADTYP
-	-	-	-	-	-	-	**0**

Bit 0

ADTYP	Resolution of A/D conversion
0	**10-bit resolution**
1	8-bit resolution

Disable interrupt.

- Interrupt mask flag registers (MK0H)
  Interrupt servicing control.

Symbol: MK0H

7	6	5	4	3	2	1	0
TMMK02	1	TMMK03H	PMK3	PMK2	KRMK	ADMK	TMMK01
	-	x			x	**1**	

Bit 1

ADMK	Interrupt servicing control
0	Interrupt servicing enabled
1	**Interrupt servicing disabled**

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

Start A/D voltage comparator operation.

- A/D converter mode register 0 (ADM0)
  A/D voltage comparator operation control.

Symbol: ADM0

7	6	5	4	3	2	1	0
ADCS	0	0	FR1	FR0	0	LV0	ADCE
	-	-			-		1

Bit 0

ADCE	A/D voltage comparator operation control
0	Stops A/D voltage comparator operation
1	**Enables A/D voltage comparator operation**

Refer to the R7F0C806-809 user's manual (hardware) for details on individual registers.

Initial values of individual bits

x: Bits not used in this application; blank spaces: bits that do not change; -: reserved bits or bits that have nothing assigned.

5.7.8 Main Processing

Figure 5.8 shows the flowchart for the main processing routine.

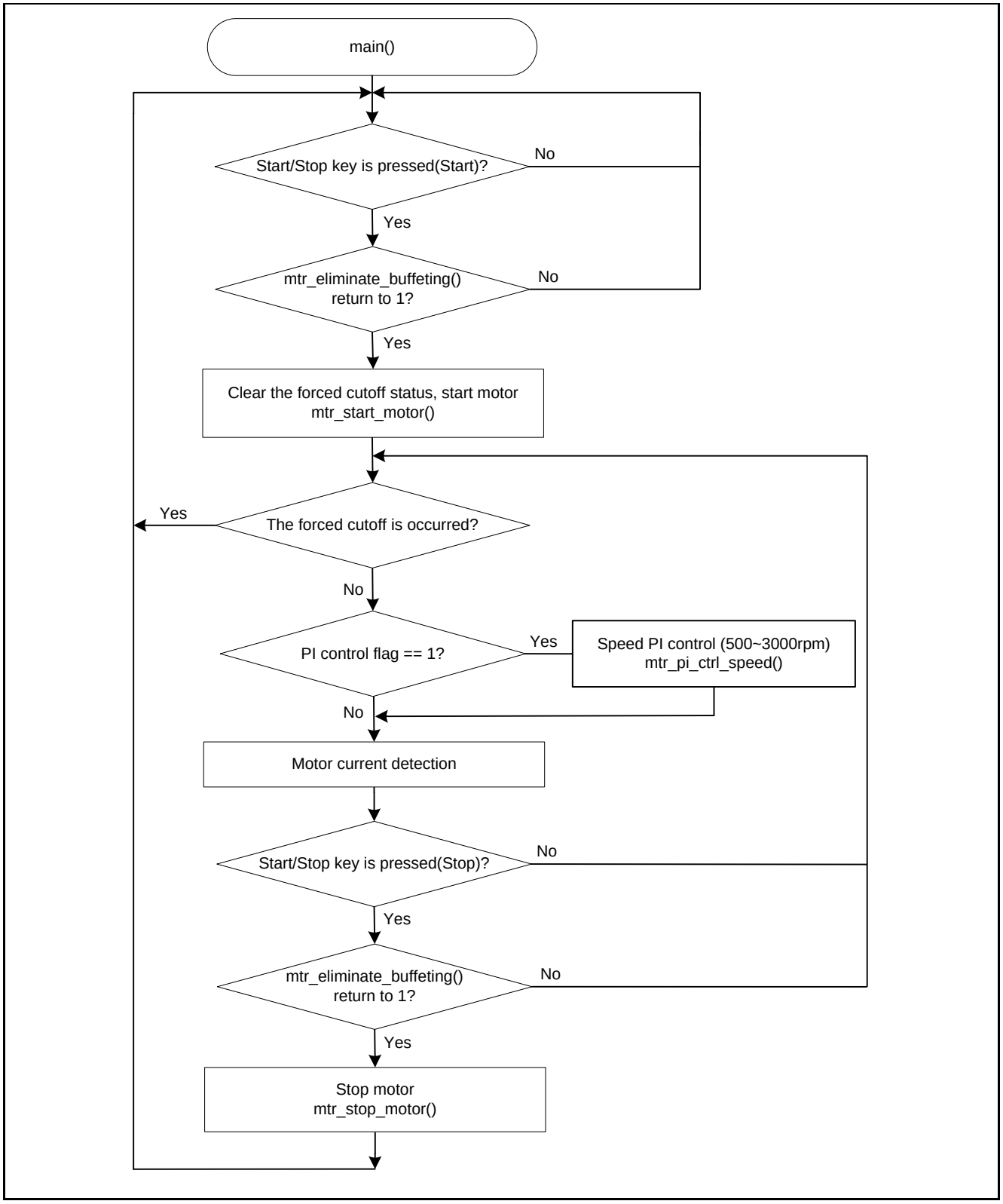

Figure 5.8 Main Processing

5.7.9 Switch Elimination Function Process

Figure 5.9 shows the flowchart for switch elimination function process.

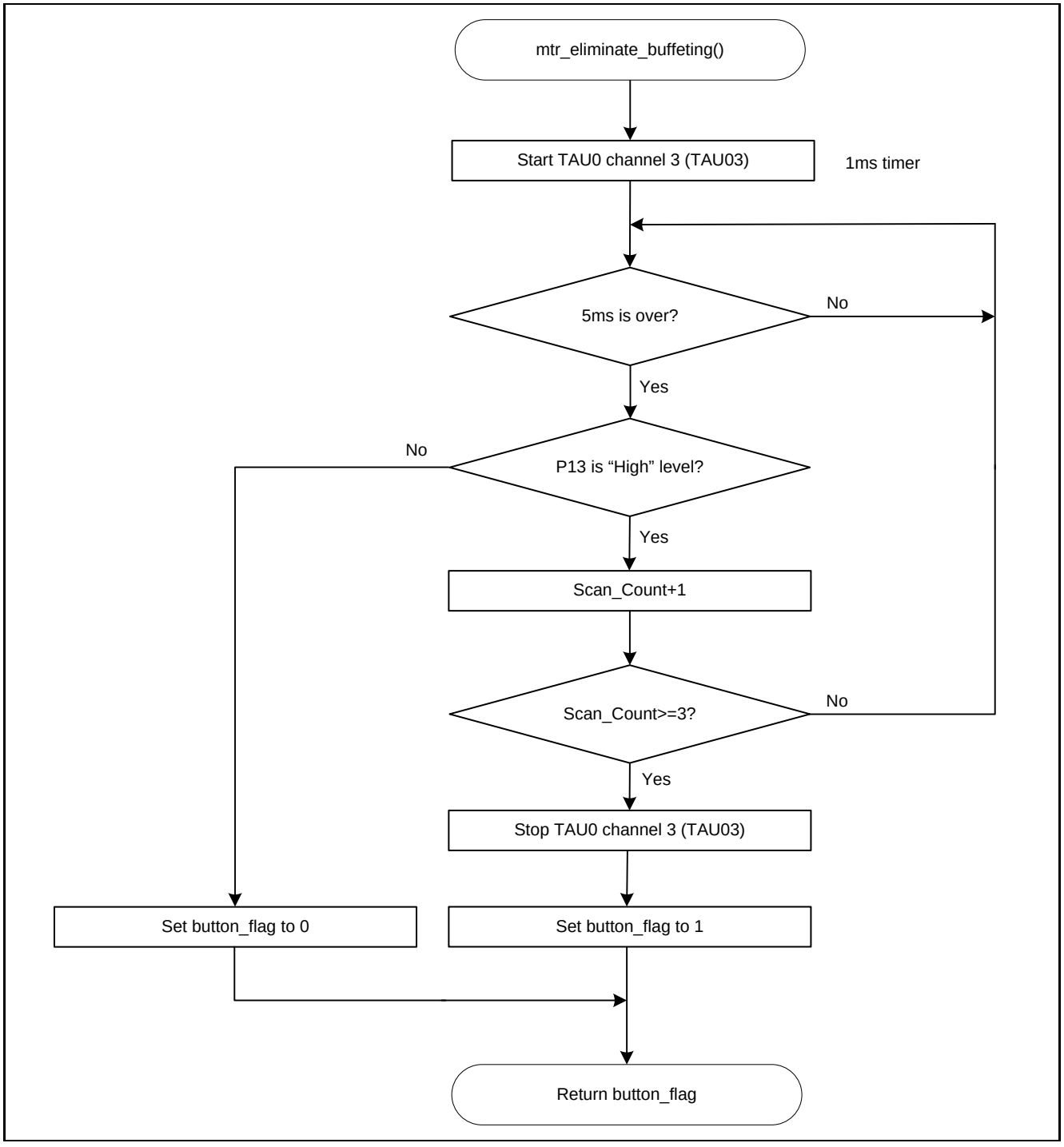

Figure 5.9 Switch Elimination Function Process

5.7.10 Motor Start Function Process

Figure 5.10 shows the flowchart for motor start function process.

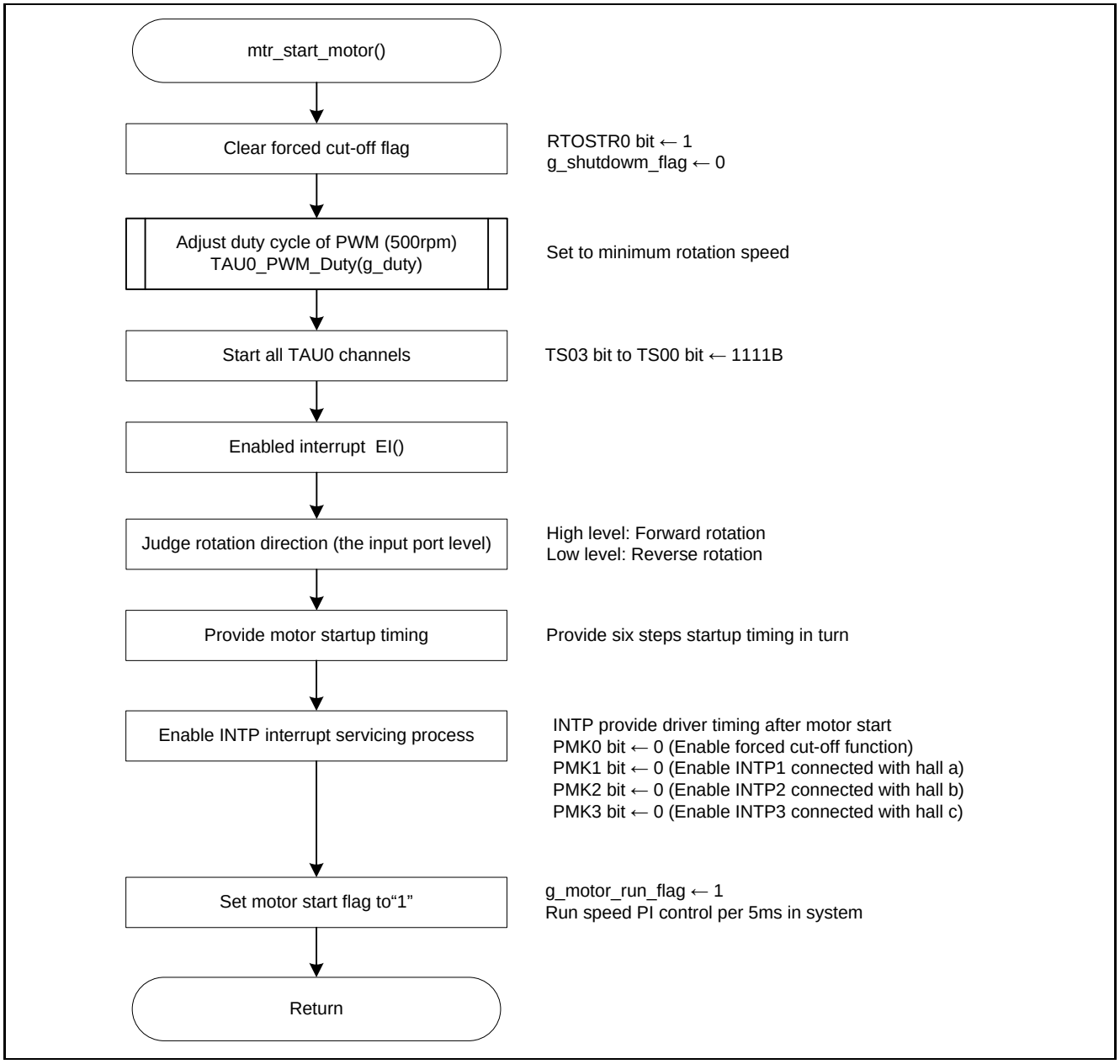

Figure 5.10 Motor Start Function Process

5.7.11 Motor Stop Function Process

Figure 5.11 shows the flowchart for motor stop function process.

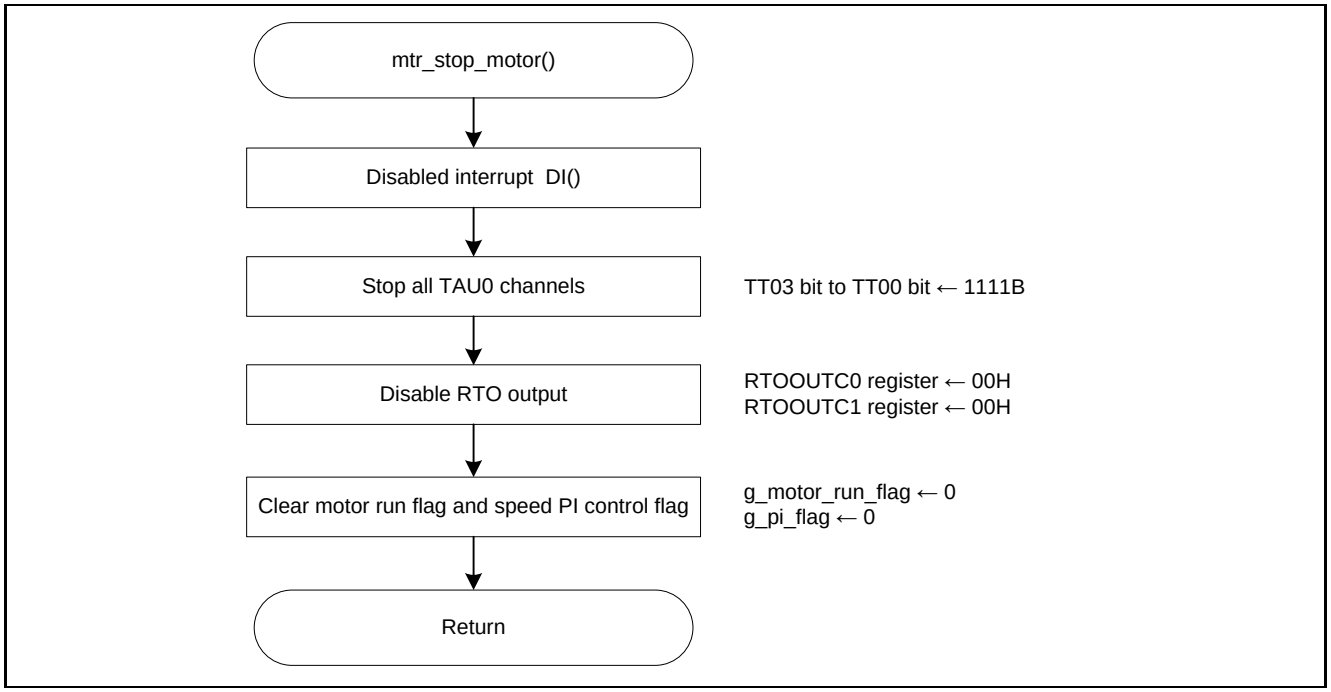

Figure 5.11 Motor Stop Function Process

5.7.12 Current Detection Function Process

Figure 5.12 shows the flowchart for Current detection function process.

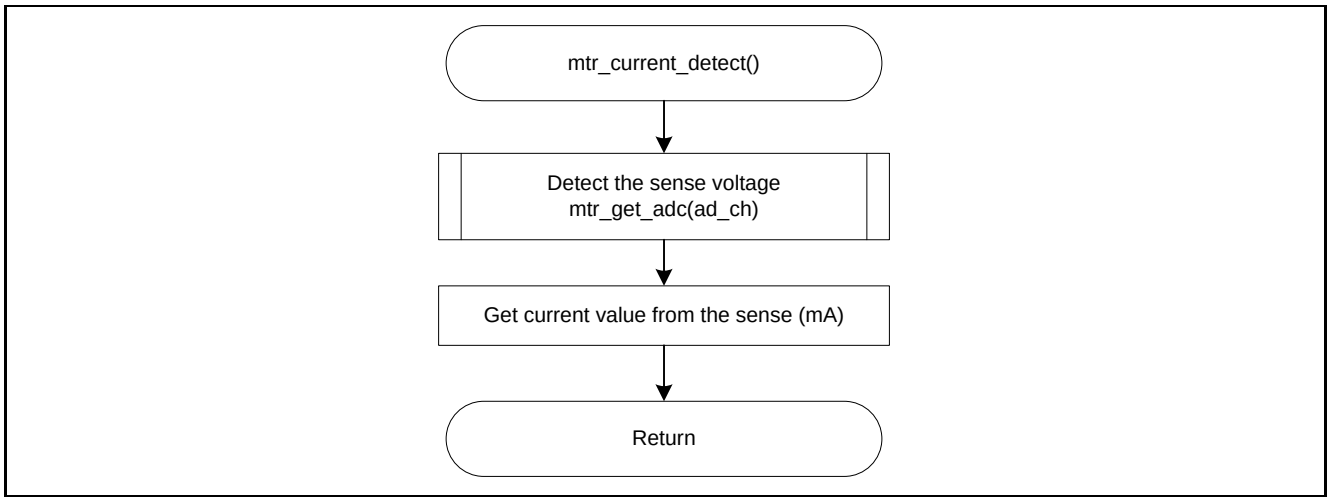

Figure 5.12 Current Detection Function Process

5.7.13 Rotation Speed Calculation Function Process

Figure 5.13 shows the flowchart for rotation speed calculation function process.

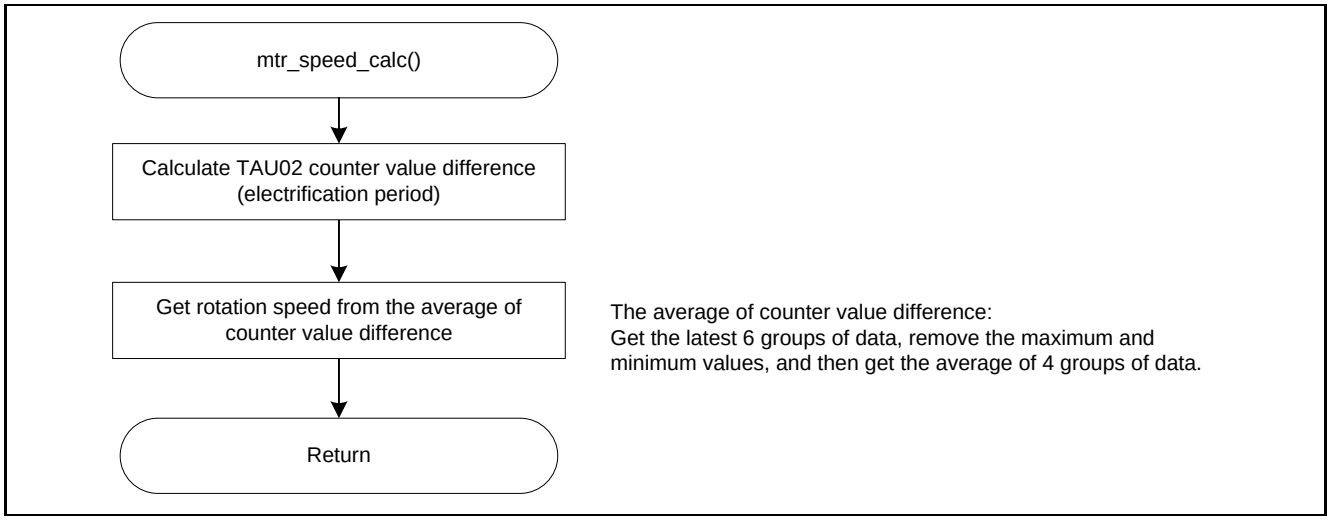

Figure 5.13 Rotation Speed Calculation Function Process

5.7.14 Speed PI Control Function Process

Figure 5.14 shows the flowchart for speed PI control function process.

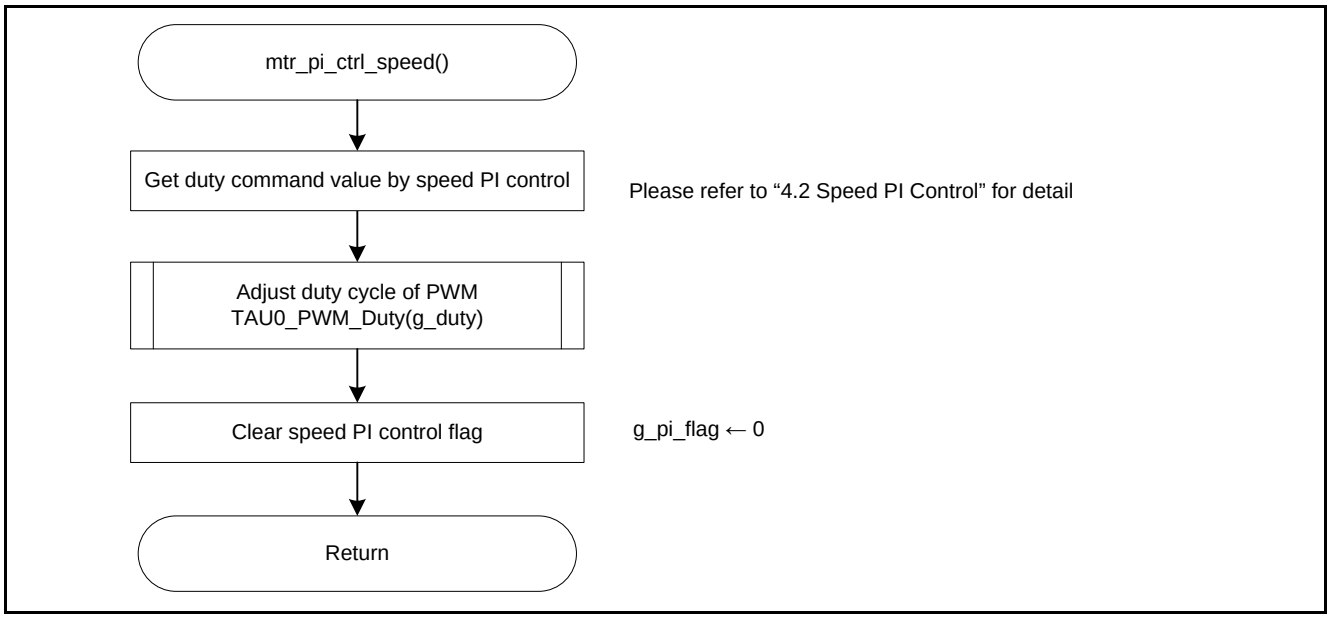

Figure 5.14 Speed PI Control Function Process

5.7.15 Interrupt Process

Figure 5.15 and figure 5.16 show the flowchart for interrupt process.

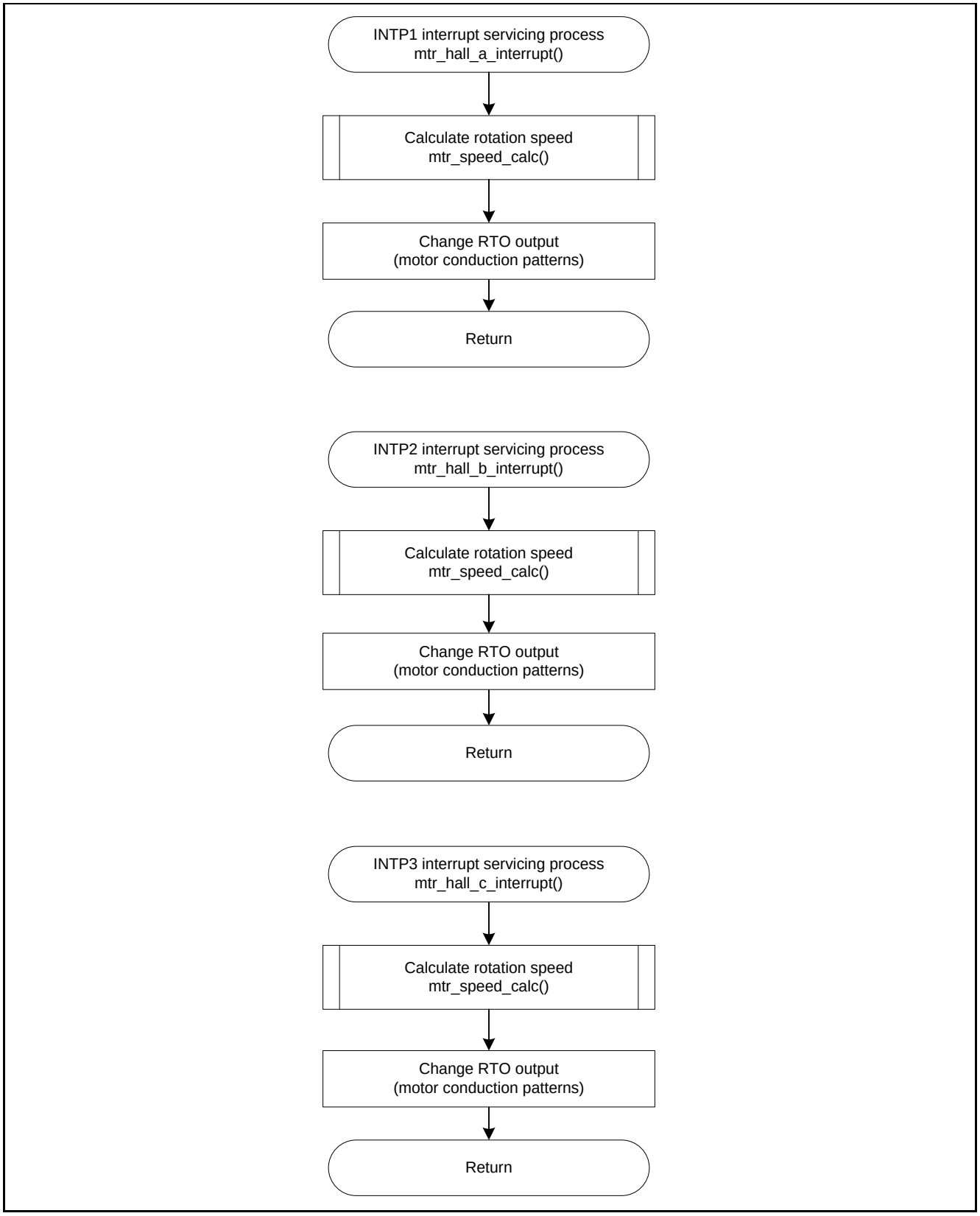

Figure 5.15 Interrupt Process (1/2)

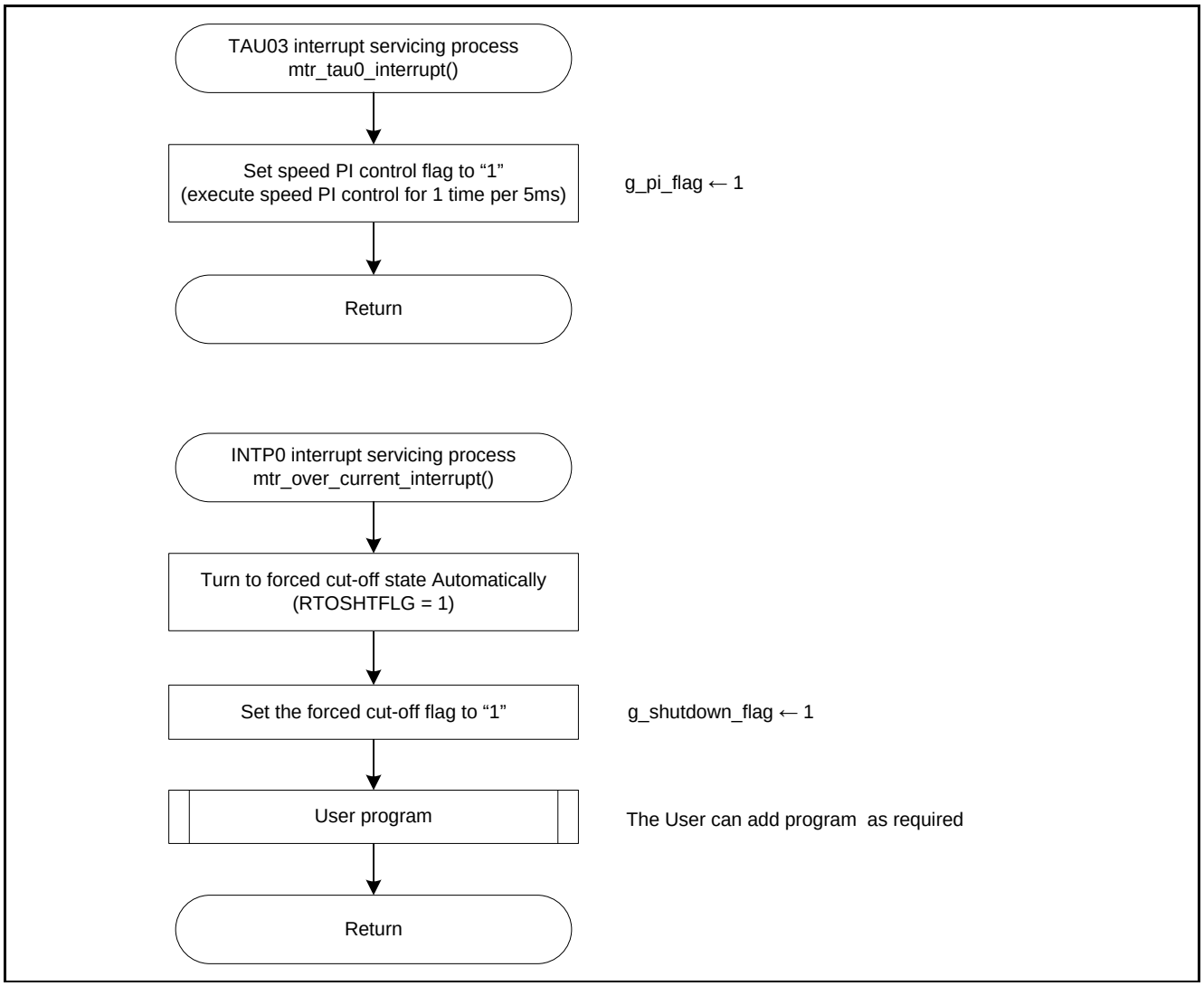

Figure 5.16 Interrupt Process (2/2)

6. Sample Code

The sample code is available on the Renesas Electronics Website.

7. Reference Documents

User's Manual

R7F0C806-809 User's Manual: Hardware (R01UH0481E)

RL78 Family User's Manual: Software (R01US0015E)

The latest versions of the documents are available on the Renesas Electronics Website.

Technical Updates/Technical News

The latest versions of the documents are available on the Renesas Electronics Website.

Website and Support

Renesas Electronics Website

http://www.renesas.com/

Inquiries

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

Revision History

		Description	
Rev.	**Date**	**Page**	**Summary**
1.00	Sep. 30, 2014	62	First edition issued

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU 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. 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 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. Unused pins should be handled as described under Handling of Unused Pins in the manual.
2. Processing at Power-on
   The state of the product is undefined at the moment 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 moment 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 moment 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 moment when power is supplied until the power reaches the level at which resetting has been specified.
3. Prohibition of Access to Reserved Addresses
   Access to reserved addresses is prohibited.
   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.
4. Clock Signals
   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has 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. Moreover, 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.
5. Differences between Products
   Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.
   — The characteristics of an MPU or MCU in the same group but having a different part number may differ in terms of the 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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of 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. 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 should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.
   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may 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 should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

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

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

RENESAS

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

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

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

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

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

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

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

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

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022/9044

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

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

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Ku, Seoul, 135-920, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 4.0