32-bit RISC Microcontroller
Reference Manual

Advanced Vector Engine 2
(A-VE2-A)

Revision 1.0

2026-02

Toshiba Electronic Devices & Storage Corporation

Contents

List of Figures

List of Tables

Preface

Notices and Legal Information

Related Document

Document name
Advanced Programmable Motor Control Circuit 2
12-bit Analog to Digital Converter
Exception
Clock Control and Operation Mode
Product Information
Secure Access Memory

Terms and Abbreviation

Some of abbreviations used in this document are as follows:

ADC	Analog to Digital Converter
PMD	Programmable Motor Control Circuit
SAM	Secure Access Memory
VE	Vector Engine

Conventions

- Numeric formats follow the rules as shown below:
  Hexadecimal: 0xABC
  Decimal: 123 or 0d123 - Only when it needs to be explicitly shown that they are decimal numbers.
  Binary: 0b111 - It is possible to omit the "0b" when the number of bits can be distinctly understood from a sentence.
- "_N" is added to the end of signal names to indicate low active signals.
- It is called "assert" that a signal moves to its active level, "deassert" to its inactive level.
- When two or more signal names are referred, they are described like as [m:n].
  Example: S[3:0] shows four signal names S3, S2, S1 and S0 together.
- The characters surrounded by ***[ ]*** defines the register.
  Example: ***[ABCD]***
- "N" substitutes suffix number of two or more same kind of registers, fields, and bit names.
  Example: ***[XYZ1]***, ***[XYZ2]***, ***[XYZ3]*** → ***[XYZn]***
- "x" substitutes suffix number or character of units and channels in the register list.
- In case of unit, "x" means A, B, and C, ...

Example: ***[ADACR0]***, ***[ADBCR0]***, ***[ADCCR0]*** → ***[ADxCR0]***

- In case of channel, "x" means 0, 1, and 2, ...
  Example: ***[T32A0RUNA]***, ***[T32A1RUNA]***, ***[T32A2RUNA]*** → ***[T32AxRUNA]***
- The bit range of a register is written like as [m: n].
  Example: Bit[3: 0] expresses the range of bit 3 to 0.
- The configuration value of a register is expressed by either the hexadecimal number or the binary number.
  Example: ***[ABCD]***<EFG> = 0x01 (hexadecimal), ***[XYZn]***<VW> = 1 (binary)
- Word and byte represent the following bit length.
  Byte: 8 bits
  Half word: 16 bits
  Word: 32 bits
  Double word: 64 bits
- Properties of each bit in a register are expressed as follows:
  R: Read only
  W: Write only
  R/W: Read and write are possible.
- Unless otherwise specified, register access supports only word access.
- The register defined as "Reserved" must not be rewritten. Moreover, do not use the read value.
- The value read from the bit having default value of "-" is unknown.
- When a register containing both of writable bits and read-only bits is written, read-only bits should be written with their default value, In the cases that default is "-", follow the definition of each register.
- Reserved bits of the write-only register should be written with their default value. In the cases that default is "-", follow the definition of each register.
- Do not use read-modified-write processing to the register of a definition which is different by writing and read out.

1. Outlines

The advanced vector engine 2 (hereafter referred to as A-VE2) can support vector control of 1 channel motor in 1 unit. The following is a list of functions.

Function classification	Function	Operation
Calculation function	Basic function	Calculate with fixed point number. Computation task for vector control Interface task for PMD (Note1) and ADC (Note2)
	Current control task	d axis PI control, q axis PI control. - Non-interference control - Output limit by voltage scalar value
	SIN/COS calculation task	Calculate sine and cosine values at phase θ - Phase interpolation and phase clipping possible
	Output voltage transformation task	- Coordinate transformation (Inverse Park transformation) - Phase transformation: 2 types (Space vector transformation, inverse Clark transformation)
	Output control task	Convert 3-phase voltage to PWM output setting of PMD (Note1) (2 types) - Output limit possibility - Compensation control of dead time
	Trigger generation task	Calculate AD conversion sampling timing set value of PMD from 3-phase duty.
	Input process task	Read the ADC (Note2) conversion results and convert it to fixed point number (2 types). - Current polarity determination (hysteresis/reverse hysteresis)
	Input current transformation task	- Phase transformation (Clark transformation) - Axis transformation (Park transformation)
	Individual function	- Arctangent (ATAN) calculation task - Square root calculation task - No operation process (NOP)
Schedule manager	Program scheduling control	- Program schedule whose task execution order and startup control can be defined - Up to 32 execution tasks can be specified
	Start control	- Repeat start - Start input schedule by end of AD conversion. Start from the standby state after the output schedule ends from the input process task by the ADC conversion end interrupt
Interrupt control	Schedule end interrupt	Interrupt that occurs when task for the END flag set to "1" is repeatedly executed the specified number of times (***[VExREPTIME]***)
	Error interrupt	Interrupt that occurs when PWM interrupt is input from PMD (Note1) during execution
Other function	Output for debugging	Outputs a signal at task transition timing during schedule operation It can be monitored by PMD's (Note1) debug output function.
	VE ROM check function	- VE ROM check function Since VE ROM can be accessed from the CPU, VE ROM can be checked using the VE ROM check function.

Note1: For details of the PMD, refer to the reference manual "Advanced Programmable Motor Control Circuit 2".

Note2: For details of the ADC, refer to the reference manual "12-bit Analog to Digital Converter".

2. Configuration

2.1. Configuration of VE

[PMDxCMPU], ***[PMDxCMPV]***, ***[PMDxCMPW]***, ***[PMDxMDOUT]***, ***[PMDxTRGSEL]***, ***[PMDxTRGCMP0]***, and ***[PMDxTRGCMP1]*** in PMD and ***[VExCMPU]***, ***[VExCMPV]***, ***[VExCMPW]***, ***[VExOUTCR]***, ***[VExTRGSEL]***, ***[VExTRGCMP0]***, and ***[VExTRGCMP1]*** in VE can be changed each other.

When ***[PMDxMODESEL]***<MDSEL[3:0]> are set to “Registers of VE are used.”, the registers in VE ae used for PMD control.

When ***[PMDxMODESEL]***<MDSEL[3:0]> are set to “Registers of PMD are used.”, the registers in PMD ae used for PMD control.

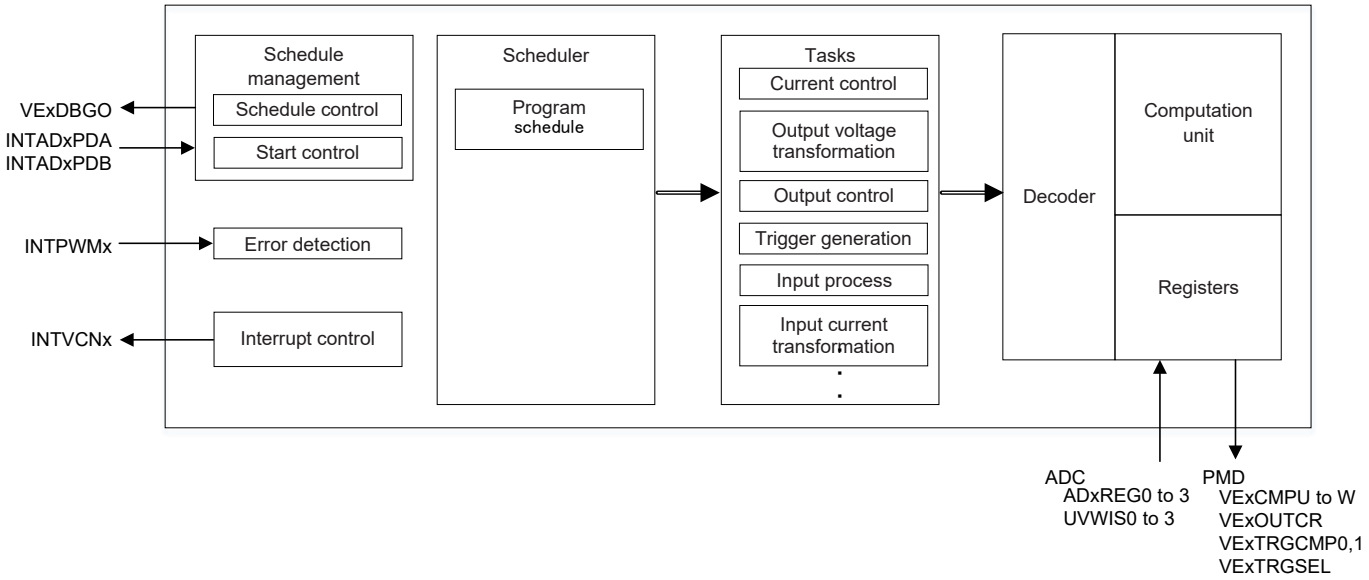


Figure 2.1 Block Diagram of VE

Table 2.1 List of Signals

No	Signal name	Signal name	I/O	Related reference manual
1	VExCMPU	U-phase PWM duty (***[VExCMPU]*** register output to PMD)	Output	Product Information
2	VExCMPV	V-phase PWM duty (***[VExCMPV]*** register output to PMD)	Output	Product Information
3	VExCMPW	W-phase PWM duty (***[VExCMPW]*** register output to PMD)	Output	Product Information
4	VExTRGCMP0	Trigger compare 0 (***[VExTRGCMP0]*** register output to PMD)	Output	Product Information
5	VExTRGCMP1	Trigger compare 1 (***[VExTRGCMP1]*** register output to PMD)	Output	Product Information
6	VExTRGSEL	Synchronous trigger output selection (***[VExTRGSEL]*** register output to PMD)	Output	Product Information
7	VExOUTCR	Conduction control / output control (***[VExOUTCR]*** register output to PMD)	Output	Product Information
8	INTPWMx	PWM interrupt (from PMD)	Input	Product Information
9	ADxREG0	ADC conversion result 0 (Current 1 data from ADC)	Input	Product Information
10	ADxREG1	ADC conversion result 1 (Current 2 data from ADC)	Input	Product Information
11	ADxREG2	ADC conversion result 2 (Current 3 data from ADC)	Input	Product Information
12	ADxREG3	ADC conversion result 3 (DC voltage data from ADC)	Input	Product Information
13	UVWIS0	Phase select information	Input	Product Information
14	UVWIS1	Phase select information	Input	Product Information
15	UVWIS2	Phase select information	Input	Product Information
16	UVWIS3	Phase select information	Input	Product Information
17	INTADxPDA	ADC conversion end interrupt A (Trigger from ADC)	Input	Product Information
18	INTADxPDB	ADC conversion end interrupt B (Trigger from ADC)	Input	Product Information
19	INTVCNx	Schedule end/error interrupt	Output	Exception, Product Information
20	VExDBGO	Task transition signal (debug output)	Output	Product Information

3. Description of Operations

3.1. Clock Supply

When you use VE, please set an applicable clock enable bit to "1" (clock supply) in Clock supply and stop register A for fsys (***[CGFSYSENA]***, ***[CGFSYSMENA]***), Clock supply and stop register B for fsys (***[CGFSYSENB]***, ***[CGFSYSMENB]***), and Clock supply and stop register for fc (***[CGFCEN]***).

An applicable register and the bit position vary according to a product. Therefore, the register may not exist with the product. Please refer to "Clock Control and Operation Mode" of the reference manual for the details.

Note: Other registers can not accessed without setting ***[VExEN]***<VEEN> to "1".

3.2. Schedule Management

Figure 3.1 shows a flowchart of motor control. The VE changes operation status according to the mode setting (***[VExMODE]***).

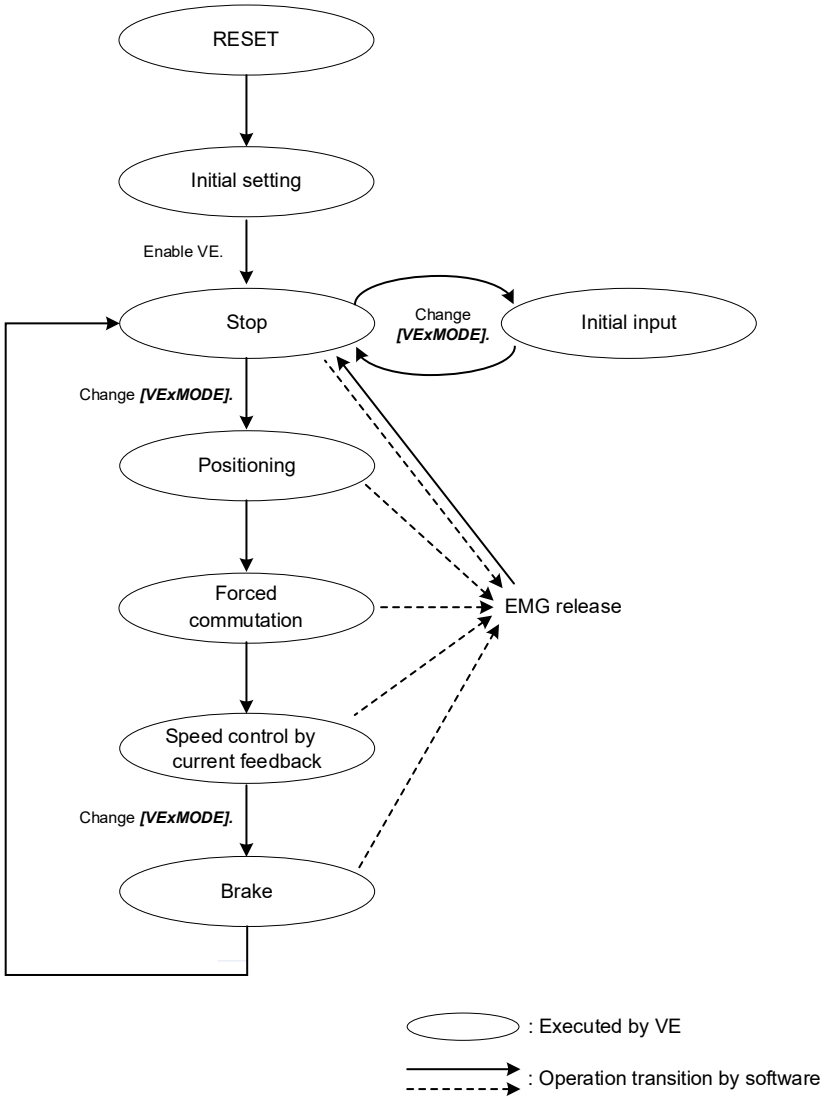


Figure 3.1 Example of Motor Control Operation Status Flowchart

Table 3.1 Motor Control Operation Status

Item	Function
RESET	Resets the MCU.
Initial setting	Specifies the initial setting by software.
Stop	Stops the motor.
Initial input	Samples and stores zero-current data when the motor is at stop.
Positioning	Controls the motor initial position.
Forced commutation	Rotates the motor. The motor is rotated without current feedback control at a specified speed in a motor activation period.
Speed control by current feedback	Control motor rotation speed by current feedback.
Brake	Deceleration control
EMG release	Release the EMG protection state.

3.2.1. Program Schedule Control

The program schedule can specify the task execution order in any combination using the program task register table setting (***[VExTSKPGMn]***) and start task specification (***[VExTSKINITP]***).

- Up to 32 execution tasks can be defined.
- ADC flag: After executing the task, VE enters ADC interrupt wait state.
- END flag: Ends the program schedule after executing the task.
- Do not set the ADC flag and END flag at the same time.
- Wraparound operation is not possible. When setting the execution task in the register ***[VExTSKPGM31]***, be sure to add the END flag.

Table 3.2 shows the example of program schedule.

Figure 3.2 shows the each execution flow when the vector engine is started on the CPU with the start task specification (***[VExTSKINITP]***) set to "0x00", "0x01", "0x06", and "0x27".

Table 3.2 Example of Program Schedule

Task	Program task registers				
		CODE	(Task name)	<ADC>	<END>
Output schedule	***[VExTSKPGM0]***	5	Current control	0	0
	[VExTSKPGM1]	6	SIN/COS calculation 1	0	0
	[VExTSKPGM2]	7	Output coordinate axis transformation (Inverse Park transformation)	0	0
	[VExTSKPGM3]	8	Output phase transformation 1 (space vector transformation)	0	0
	[VExTSKPGM4]	0	Output control 1	0	0
	[VExTSKPGM5]	1	Trigger generation	1	0
Input schedule	***[VExTSKPGM6]***	2	Input process 1	0	0
	[VExTSKPGM7]	3	Input phase transformation (Clark transformation)	0	0
	[VExTSKPGM8]	4	Input coordinate axis transformation (Park transformation)	0	1
Output schedule	***[VExTSKPGM9]***	5	Current control	0	0
	[VExTSKPGM10]	6	SIN/COS calculation 1	0	0
	[VExTSKPGM11]	7	Output coordinate axis transformation (Inverse Park transformation)	0	0
	[VExTSKPGM12]	8	Output phase transformation 1 (space vector transformation)	0	0
	[VExTSKPGM13]	0	Output control 1	0	0
	[VExTSKPGM14]	1	Trigger generation	1	0
Input schedule	***[VExTSKPGM15]***	10	Input process 2	0	0
	[VExTSKPGM16]	3	Input phase transformation (Clark transformation)	0	0
	[VExTSKPGM17]	4	Input coordinate axis transformation (Park transformation)	0	1
Output schedule	***[VExTSKPGM18]***	5	Current control	0	0
	[VExTSKPGM19]	6	SIN/COS calculation 1	0	0
	[VExTSKPGM20]	7	Output coordinate axis transformation (Inverse Park transformation)	0	0
	[VExTSKPGM21]	8	Output phase transformation 1 (space vector transformation)	0	0
	[VExTSKPGM22]	9	Output control 2	0	0
	[VExTSKPGM23]	1	Trigger generation	1	0
Input schedule	***[VExTSKPGM24]***	10	Input process 2	0	0
	[VExTSKPGM25]	3	Input phase transformation (Clark transformation)	0	0
	[VExTSKPGM26]	4	Input coordinate axis transformation (Park transformation)	0	1
Individual task execution	***[VExTSKPGM27]***	8	Output phase transformation 1 (space vector transformation)	0	1

Task		Program task registers			
		CODE	(Task name)	<ADC>	<END>
(Unused)	***[VExTSKPGM28]***	-	-	-	-
	[VExTSKPGM29]	-	-	-	-
	[VExTSKPGM30]	-	-	-	-
	[VExTSKPGM31]	-	-	-	-

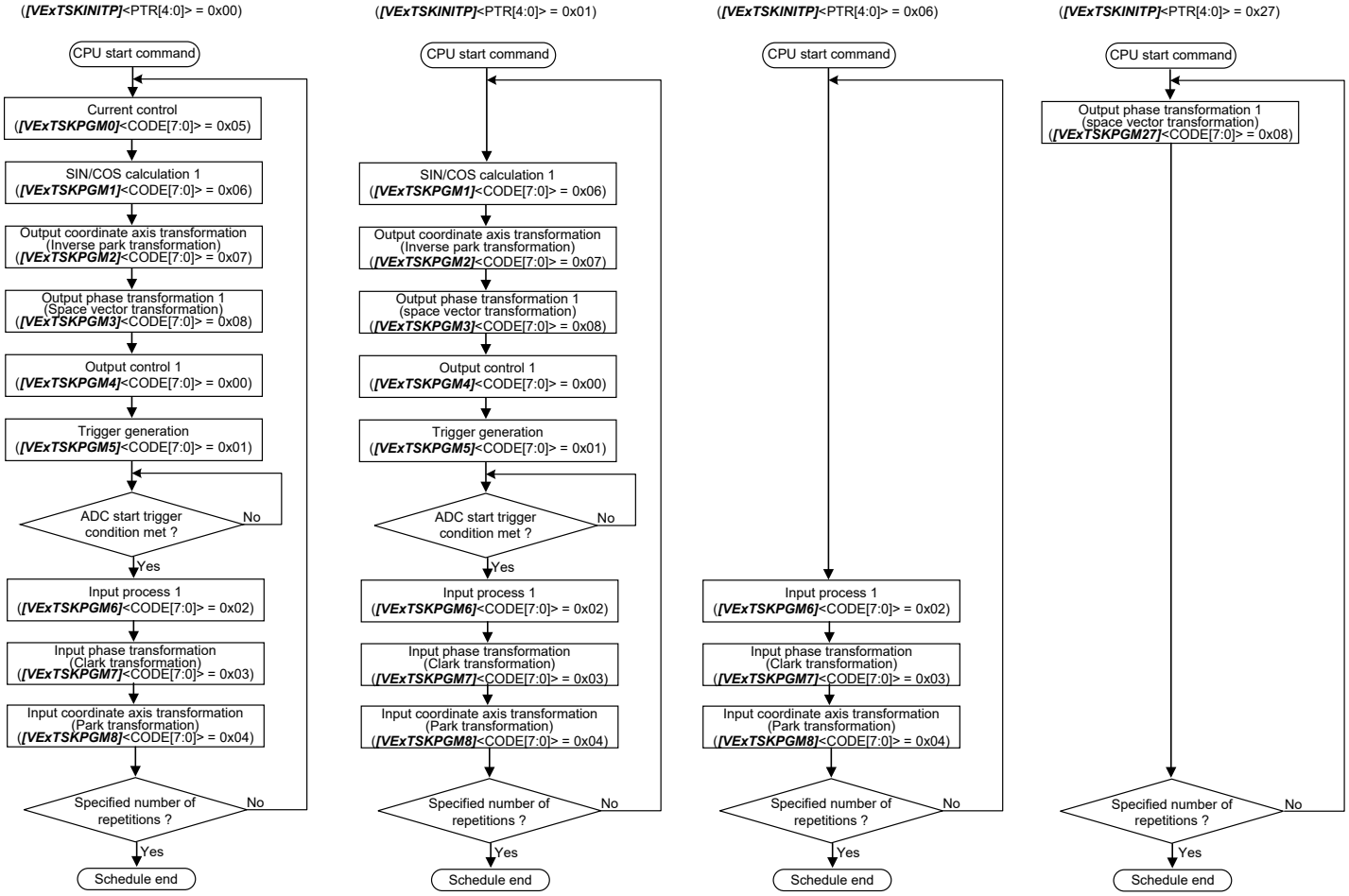


Figure 3.2 Example of Program Schedule Execution Flow

Example of timing chart for task control is shown below.

For example, indicates when ADC sampling is performed in synchronization with the basic carrier.

- Timing of ADC trigger settings (1) and (2) is determined by VE's ***[VExTRGCMPn]*** (n=0,1).
- Timing of ADC trigger setting (3) is set by user for ***[PMDxTRGCR]*** of PMD.
- Information obtained in ADC trigger settings (1) to (3), shown as gray hatting, is used in the processing shown as gray hatting of the next PWM cycle in the below figure.
- Execute the output schedule so that it is completed within the PWM cycle. If PWM interrupt occurs before completion, error interrupt can be generated, so use if necessary.
- VE control cycle starts from the output schedule and ends at the completion of VE interrupt process.

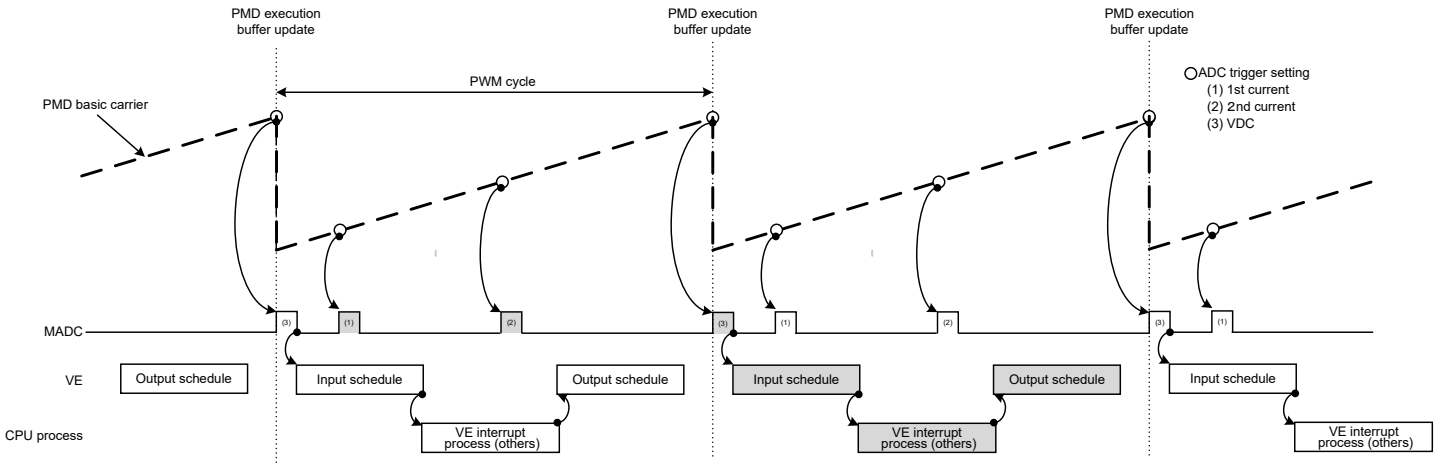


Figure 3.3 Example of Timing Chart for Task Control

3.2.2. Start Control

First, enable VE (***[VExEN]***<VEEN> = 1) and set the program task register (***[VExTSKPGMn]***), program task pointer start register (***[VExTSKINITP]***), start trigger mode setting register (***[VExTRGMODE]***) and operation schedule repeat count specification register (***[VExREPTIME]***). Then start VE operation by CPU start trigger selection register (***[VExCPURUNTRG]***).

Table 3.3 Schedule-Related Registers

Register	Function	
[VExTSKPGMn]	Execution task order specification	Specify tasks in execution order.
[VExTSKINITP]	Star task specification	Set start task.
[VExREPTIME]	Schedule repeat number	Set from "1" to "15". (Note) Set to "1" even for one-time execution. Schedule cannot be executed with "0".
[VExTRGMODE]	Start trigger mode setting	Input schedule trigger selection INTADxPDA or INTADxPDB interrupt
[VExCPURUNTRG]	VE start	Operation start register

3.2.3. Interrupt Control

VE has schedule end interrupt (INTVCNx) that occurs when the program schedule ends.

- Schedule end interrupt
  INTVCNx interrupt occurs when task with the END flag set to "1" is repeatedly executed the specified number of times (***[VExREPTIME]***) (at the end of execution).

- Error interrupt
  When the error detection interrupt is enabled (***[VExERRINTEN]*** = 1), INTVCNx interrupt occurs and the error flag (***[VExERRDET]***) is set to "1" when PWM interrupt of the PMD circuit occurs during program schedule execution.

3.3. Description of Tasks

This subsection describes the outline of each task operation in the schedule.
Table 3.4 shows the task numbers that are used to specify the execution task and the start task.

Table 3.4 List of Tasks

Task		Task function	Task number
Output schedule	Current control	PI control for d-axis/q-axis (PI control output limit enable) Non-interference control for d-axis and q-axis, voltage scalar limitation	5
	SIN/COS calculation	Sine/cosine calculation Phase interpolation (with clipping function)	6
	Output coordinate axis transformation	Inverse Park transformation	7
	Output phase transformation 1	Transforms from 2-phase to 3-phase [SVM]	8
	Output phase transformation 2	Transforms from 2-phase to 3-phase [Inverse Clark transformation]	11
	Output control 1	Converts the data to PMD setting format. Switches PWM shift 1. Limits the PWM output. Sets the dead time compensation control.	0
	Output control 2	Converts the data to PMD setting format. Switches PWM shift 2. Limits the PWM output. Sets the dead time compensation control.	9
	Trigger generation	Generates the synchronous trigger timing for ADC.	1
Input schedule	Input process 1	Corresponds to sensor, 3-shunt and 1-shunt current detection.1-shunt is corresponded in the case of PWM shift prohibited or PWM shift 1. Captures the result of AD conversion and converts them into fixed-point format. Specifies the hysteresis width to determine the current polarity.	2
	Input process 2	Corresponds to sensor, 3-shunt and 1-shunt current detection.1-shunt is corresponded in the case of PWM shift 2. Captures the result of AD conversion and converts them into fixed-point format. Specifies the hysteresis width to determine the current polarity.	10
	Input phase transformation	Transforms from 3-phase to 2-phase.	3
	Input coordinate axis transformation	Performs Park transformation Calculates the declination angle of the current vector or the induced voltage vector on the d-q coordinate.	4
ATAN2 calculation		Calculates the arctangent.	12
SQRT calculation		Calculates the square root.	13
NOP		NOP is used for pausing process and waiting until AD conversion is completed after output control.	239

Table 3.5 shows which input process tasks and output control tasks correspond to each combination of PWM shift methods and current detection methods.

Table 3.5 Correspondence Table of Input Process Task, Output Control Task and PWM Shift, Current Detection Metods

Current detection method		Input process 1 task	Input process 2 task	Output control 1 task	Output control 2 task
Shunt resistance number	PWM shift				
1-shunt	Shift 1	✓	-	✓	-
1-shunt	Shift 2	-	✓	-	✓
1-shunt	Normal 2-phase modulation	✓	-	✓	-
1-shunt	Normal 3-phase modulation	✓	-	✓	-
3-shunt	any	✓	-	✓	-

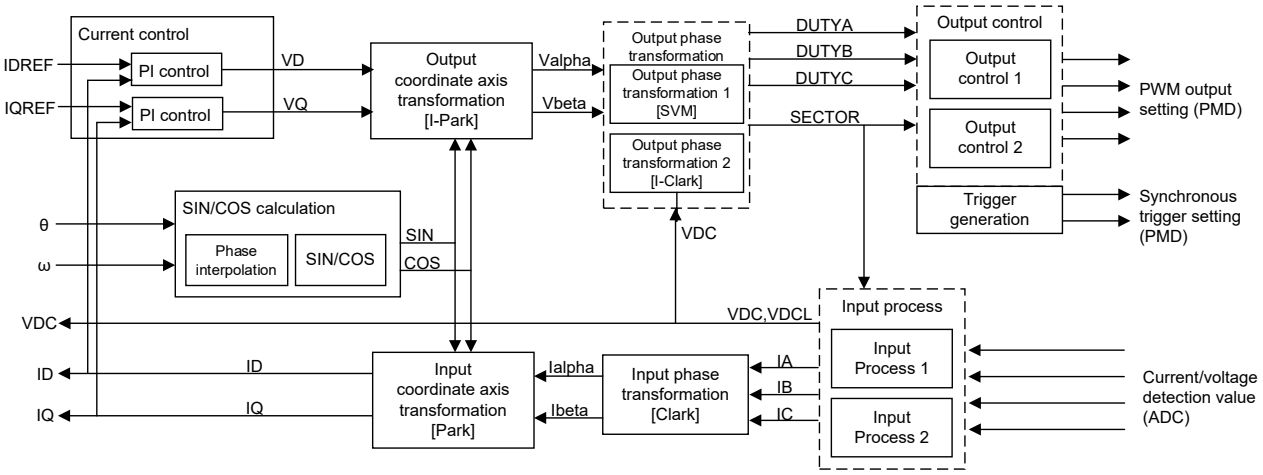


Figure 3.4 Relationship Diagram of Tasks

3.3.1. Current Control (Task 5)

The current control task is comprised of a PI control unit for d-axis current and a PI control unit for q-axis current, and calculates the d-axis and q-axis voltages respectively.

The expansion control enables the non-interference control and the voltage scalar limitation that controls d-axis and q-axis together.

a. d-axis current PI control

<Equation>

[PI control]

Δid = ***[VExIDREF] - [VExID]*** ; Calculates between the current reference value and the current feedback

kpg = (***[VExCIDKG]***<CIDKPG> setting) ; Range of proportional coefficient

kig = (***[VExCIDKG]***<CIDKIG> setting) ; Range of integral coefficient

limit = ***[VExPIOLIM]***

if (kpg ≥ 2) n = kpg ; Selected range is more than twice Save range of proportional coefficient

limit = limit / kpg ; Limit correction

kig = kig / kpg ; Range correction of integral

kpg = 1 ; Range correction of proportional

else n = 1

cidkp = ***[VExCIDKP]*** × kpg ; Proportional coefficient

cidki = ***[VExCIDKI]*** × kig ; Integral coefficient

vdi0 = cidki × Δid + VDI ; Calculates the integral components.

vd0 = cidkp × Δid + vdi0 ; Calculates the voltage by adding the proportional components.

[PI control output limitation]

if (vd0 > limit) ; The upper limit value

vd = limit

[VExMCTLF]<PIDOVF> = 1

else if (vd0 < -limit) ; The lower limit value

vd = -limit

[VExMCTLF]<PIDOVF> = 1

else vd = vd0

[VExVD] = vd × n ; Save more than a double proportional range to be compensated.

[Anti-windup]

Δvd = vd - vd0 ; Calculates the difference between the d-axis voltage and the limit.

VDI = vdi0 + Δvd × (***[VExMODE]***<AWUMD> setting) ; Reflects the above difference to the integral component.

	Register name/ signal name	Function	Detail
Input	***[VExID]***	d-axis current	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExIDREF]	d-axis current reference setting	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExCIDKP]	d-axis current proportional coefficient setting	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExCIDKI]	q-axis current integral coefficient setting	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExCIDKG]	[3:0] Selects a PI control d-axis integral coefficient range.	<CIDKIG[3:0]> Selects a PI control d-axis integral coefficient range. 0000: 1 / 1 0001: 1 / 2^4 0010: 1 / 2^8 0011: 1 / 2^{12} 0100: 1 / 2^{16} 0101 to 1111: Reserved
		[7:4] Selects a PI control d-axis proportional coefficient range.	<CIDKPG[3:0]> Selects a PI control d-axis proportional coefficient range. 0000: 1 / 1 0001: 1 / 2^4 0010: 1 / 2^8 0011: 1 / 2^{12} 0100: 1 / 2^{16} 0101 to 1000: Reserved 1001: 2 1010: 2^2 1011: 2^3 1100: 2^4 1101 to 1111: Reserved
	[VExPIOLIM]	PI control output limit value setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits) Valid range: "0x0000" to "0x7FFF" Output limitation is disabled when ***[VExPIOLIM]*** = 0x0000.
	[VExMODE]	[9:8] Specifies anti-windup (AWU) control when PI control output limitation is performed.	<AWUMD[1:0]> 00: AWU control is disabled. 01: Substitutes the result from amount of limitation / 4 into the integral term. 10: Substitutes the result from amount of limitation / 2 into the integral term. 11: Substitutes the amount of limitation into the integral term.
	VDI	d-axis voltage integral component store	64-bit fixed-point data (-1.0 to 1.0, 63 fractional bits)
Output	***[VExVD]***	d-axis voltage	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExMCTLF]	[8] Indicates the excess flag for d-axis output limitation on PI control	<PIDOVF> 0: |d-axis output on PI control| ≤ ***[VExPIOLIM]*** 1: |d-axis output on PI control| > ***[VExPIOLIM]***
	VDI	d-axis voltage integral component store	64-bit fixed-point data (-1.0 to 1.0, 63 fractional bits)

Note: The VDI is comprised of 64 bits. The upper is used for the ***[VExVDIH]*** register and the lower is used for the ***[VExVDILH]*** register

b. q-axis current PI control

<Equation>

[PI control]

Δiq = ***[VExIQREF] - [VExIQ]*** ; Calculates between the current reference value and the current feedback

kpg = (***[VExCIQKG]***<CIQKPG> setting) ; Range of proportional coefficient

kig = (***[VExCIQKG]***<CIQKIG> setting) ; Range of integral coefficient

limit = ***[VExPIOLIM]***

if (kpg ≥ 2) n = kpg ; Selected range is more than twice, Save range of proportional coefficient

limit = limit / kpg ; Limit correction

kig = kig / kpg ; Range correction of integral

```
                kpg = 1                          ; Range correction of proportional
else            n = 1
ciqkp = [VExCIQKP] × kpg                         ; Proportional coefficient
ciqki = [VExCIQKI] × kig                         ; Integral coefficient
vqi0 = ciqki × Δiq + VQI                         ; Calculates the integral components.
vq0 = ciqkp × Δiq + vqi0                         ; Calculates the voltage by adding the
                                                   proportional components.

[PI control output limitation]
if (vq0 > limit)                                 ; The upper limit value
        vq = limit
        [VExMCTLF]<PIQOVF> = 1
else if (vq0 < -limit)                           ; The lower limit value
        vq = -limit
        [VExMCTLF]<PIQOVF> = 1
else    vq = vq0
[VExVQ] = vq × n                                 ; Save more than a doubled proportional range
                                                   to be compensated.

[Anti-windup]
Δvq = vq - vq0                                   ; Calculates the difference between the q-axis
                                                   voltage and the limit.
VQI = vqi0 + Δvq × ([VExMODE]<AWUMD> setting)    ; Reflects the above difference to the
                                                   integral component.
```

	Register name/ signal name	Function	Detail
Input	***[VExIQ]***	q-axis current	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExIQREF]	q-axis current reference setting	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExCIQKP]	q-axis current proportional coefficient setting	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExCIQKI]	q-axis current integral coefficient setting for PI Control	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExCIQKG]	[3:0] Selects a PI control q-axis integral coefficient range.	<CIQKIG[3:0]> Selects a PI control q-axis integral coefficient range. 0000: 1 / 1 0001: 1 / 2^4 0010: 1 / 2^8 0011: 1 / 2^{12} 0100: 1 / 2^{16} 0101 to 1111: Reserved
		[7:4] Selects a PI control q-axis proportional coefficient range.	<CIQKPG[3:0]> Selects a PI control q-axis proportional coefficient range. 0000: 1 / 1 0001: 1 / 2^4 0010: 1 / 2^8 0011: 1 / 2^{12} 0100: 1 / 2^{16} 0101 to 1000: Reserved 1001: 2 1010: 2^2 1011: 2^3 1100: 2^4 1101 to 1111: Reserved
	[VExPIOLIM]	PI control output limit value setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits) Valid range: "0x0000" to "0x7FFF" Output limitation is disabled when ***[VExPIOLIM]*** = 0x0000.

	Register name/ signal name	Function	Detail
	[VExMODE]	[9:8] Specifies anti-windup (AWU) control when PI control output limitation is performed.	<AWUMD[1:0]> 00: AWU control is disabled. 01: Substitutes the result from amount of limitation / 4 into the integral term. 10: Substitutes the result from amount of limitation / 2 into the integral term. 11: Substitutes the amount of limitation into the integral term.
	VQI	q-axis voltage integral component store	64-bit fixed-point data (-1.0 to 1.0, 63 fractional bits)
Output	***[VExVQ]***	q-axis voltage	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExMCTLF]	[9] Indicates the excess flag for q-axis output limitation on PI control.	<PIQOVF> 0: |q-axis output on PI control | ≤ ***[VExPIOLIM]*** 1: |q-axis output on PI control | > ***[VExPIOLIM]***
	VQI	q-axis voltage integral component store	64-bit fixed-point data (-1.0 to 1.0, 63 fractional bits)

Note: The VQI is comprised of 64 bits. The upper is used for the ***[VExVQIH]*** register and the lower is used for the ***[VExVQILH]*** register

Note: ***[VExTMPREG0]*** is updated to an undefined value because it is used to temporarily save internal data during task execution.

c. Non-interference control

The result of PI control is corrected using the results of the interference to d-axis and q-axis based on motor voltage equation.

<Equation>

```
if ([VExMODE]<T5ECEN> = 1)                                  ; Expansion control is enabled
   ld = [VExCLD] × ([VExCLG] setting)                       ; d-axis inductance
   lq = [VExCLQ] × ([VExCLG] setting)                       ; q-axis inductance
   phi = [VExCPHI] × ([VExCPHIG] setting)                   ; Interlinkage magnetic flux
   id = [VExID]                                             ; Feedback current
   iq = [VExIQ]
   if ([VExFMODE]<IDQSEL> = 1)   id = [VExIDREF]            ; Inputs a command value
                                 iq = [VExIQREF]
   [VExVDE] = - [VExOMEGA] × iq × lq                        ; Calculates the interference to d-axis
   [VExVQE] = [VExOMEGA] × id × ld + [VExOMEGA] × phi       ; Calculates the interference to q-axis
   if ([VExMODE]<NICEN> = 1)                                ; Non-interference control is enabled
      [VExVD] = [VExVD] + [VExVDE]
      [VExVQ] = [VExVQ] + [VExVQE]
```

	Register name	Function	Detail
Input	***[VExVD]***	d-axis voltage	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExVQ]	q-axis voltage	
	[VExID]	d-axis current	
	[VExIQ]	q-axis current	
	[VExIQREF]	q-axis current reference setting	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExIDREF]	d-axis current reference setting	
	[VExCLD]	Motor d-axis inductance value	16-bit fixed-point data (-16 to 16, 11 fractional bits)

	Register name	Function	Detail
	[VExCLQ]	Motor q-axis inductance value	
	[VExCPHI]	Motor interlinkage magnetic flux	
	[VExCLG]	Motor inductance range setting	<CLG[2:0]>, <CPHIG[2:0]> 000: 1 / 1 001: 1 / 2^4 010: 1 / 2^8 011: 1 / 2^{12} 100: 1 / 2^{16} 101 to 111: Reserved
	[VExCPHIG]	Motor magnetic flux range setting	
	[VExOMEGA]	Rotational speed setting	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExFMODE]	[4] Selects non-interference control input current.	<IDQSEL> 0: Feedback current (***[VExID]***, ***[VExIQ]***) 1: Command current (***[VExIDREF]***, ***[VExIQREF]***)
	[VExMODE]	[11] Disables/enables non-interference control	<NICEN> 0: Disabled 1: Enabled
		[10] Disables/enables expansion control (non-interference control and voltage scalar limitation)	<T5ECEN> 0: Disabled 1: Enabled
Output	***[VExVDE]***	Indicates the value of d-axis calculation of non-interference	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExVQE]	Indicates the value of q-axis calculation of non-interference	
	[VExVD]	d-axis voltage	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExVQ]	q-axis voltage	

d. Voltage scalar limitation

The voltage scalar limitation controls d-axis voltage and q-axis voltage, so as to be lower the composite value (the square root of $VD^2 + VQ^2$) that is comprised of d-axis voltage and q-axis voltage than the limit value.

<Equation>

if (***[VExMODE]***<T5ECEN> = 1) ; Expansion control is enabled.
[VDQ calculation]
if ((***[VExVD]***^{2} + ***[VExVQ]***^{2})> ***[VExVSLIM]***^{2}) ; Confirm the excess
if (***[VExFMODE]***<VSLIMMD[1:0]> = 00) ; Voltage scalar limitation is disabled
[VExVDQ] = SQRT (***[VExVD]***^{2} + ***[VExVQ]***^{2})
else if (***[VExFMODE]***<VSLIMMD[1:0]> = 01) ; Scalar limitation on the d-axis direction
[VExVDQ] = SQRT (***[VExVSLIM]***^{2} - ***[VExVQ]***^{2})
else if (***[VExFMODE]***<VSLIMMD[1:0]> = 10) ; Scalar limitation on the q-axis direction
[VExVDQ] = SQRT (***[VExVSLIM]***^{2} - ***[VExVD]***^{2})
else if (***[VExFMODE]***<VSLIMMD[1:0]> = 11) ; dq proportional scalar limitation
[VExVDQ] = SQRT (***[VExVD]***^{2} + ***[VExVQ]***^{2})
else ***[VExVDQ]*** = SQRT (***[VExVD]***^{2} + ***[VExVQ]***^{2}) ; Calculation of voltage scalar value
Note: SQRT is the square root calculation

[Calculation for the declination angle] *When ***[VExVSLIM]*** = 0, disable the process of calculating the declination angle.

x = |***[VExVQ]***|

y = |***[VExVD]***|
[VExVDELTA] = ATAN (x, y)
Note: ATAN is arctangent calculation

[Limitation calculation for each axis] *When ***[VExVSLIM]*** = 0, disable the process of calculating each axis limit value.

if (***[VExFMODE]***<VSLIMMD[1:0]> = 00) ; Voltage scalar limitation is disabled
vdlim = ***[VExVSLIM]***
vqlim = ***[VExVSLIM]***
else if (***[VExFMODE]***<VSLIMMD[1:0]> = 01) ; Scalar limitation on the d-axis direction
vdlim = ***[VExVDQ]***
vqlim = ***[VExVSLIM]***
else if (***[VExFMODE]***<VSLIMMD[1:0]> = 10) ; Scalar limitation on the q-axis direction
vdlim = ***[VExVSLIM]***
vqlim = ***[VExVDQ]***
else if (***[VExFMODE]***<VSLIMMD[1:0]> = 11) ; dq proportional scalar limitation
vdlim = ***[VExVSLIM]*** × ***S***IN (***[VExVDELTA]***)
vqlim = ***[VExVSLIM]*** × COS (***[VExVDELTA]***)

[Limitation process] *When ***[VExVSLIM]*** = 0, disable limitation process.

if (***[VExVD]*** > vdlim)
[VExVD] = vdlim ; Process for the upper limit of d-axis
[VExMCTLF]<VSOVF> = 1
else if (***[VExVD]*** < -vdlim)
[VExVD] = -vdlim ; Process for the lower limit of d-axis
[VExMCTLF]<VSOVF> = 1
if (***[VExVQ]*** > vqlim)
[VExVQ] = vqlim ; Process for the upper limit of q-axis
[VExMCTLF]<VSOVF> = 1
else if (***[VExVQ]*** < -vqlim)
[VExVQ] = -vqlim ; Process for the lower limit of q-axis
[VExMCTLF]<VSOVF> = 1

[Correction process] *When ***[VExVSLIM]*** = 0, disable correction process.

[VExVD] = ***[VExVD]*** + ***[VExVDCRC]*** ; d-axis correction
[VExVQ] = ***[VExVQ]*** + ***[VExVQCRC]*** ; q-axis correction

	Register name	Function	Detail
Input	***[VExVD]***	d-axis voltage	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExVQ]	q-axis voltage	
	[VExVSLIM]	Voltage scalar limitation	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits) "0x0000" to "0x7FFF" Limitation is disabled when ***[VExVSLIM]*** = 0x0000
	[VExVDCRC]	d-axis voltage correction	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExVQCRC]	q-axis voltage correction	

	Register name	Function	Detail
	[VExMODE]	[10] Disables/enables expansion control (Non-interference control and voltage scalar limitation).	<T5ECEN> 0: Disabled 1: Enabled
	[VExFMODE]	[11:10] Controls voltage scalar limitation of current control.	<VSLIMMD[1:0]> 00: Scalar limitation is disabled. (Limitation in each axis is enabled.) 01: Limits the voltage on the d-axis. 10: Limits the voltage on the q-axis. 11: dq proportional limitation
Output	***[VExVDQ]***	Voltage scalar	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExVDELTA]	Voltage declination angle	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits) "0x0000" to "0x4000" (0 to 90 degrees)
	[VExVD]	d-axis voltage	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExVQ]	q-axis voltage	
	[VExMCTLF]	[10] Indicates the excess flag for voltage scalar limitation.	<VSOVF> 0: Voltage scalar ≤ ***[VExVSLIM]*** 1: Voltage scalar > ***[VExVSLIM]***

Note: ***[VExTMPREG0]*** is updated to an undefined value because it is used to temporarily save internal data during task execution.

3.3.2. SIN/COS Calculation (Task 6)

The SIN/COS calculation task executes phase interpolation calculation and SIN/COS calculation.

Phase interpolation calculates the rotation speed by integrating with the PWM period. It is executed only when phase interpolation is enabled (***[VExMODE]***<PVIEN> = 1).

Also, when phase interpolation is prohibited (***[VExMODE]*** = 0), theta0 = ***[VExTHETA]*** is executed.

The clipping function is a function that limits the phase interpolation so that it does not over-interpolate when dynamically reversing the motor in a sensor-equipped system. Clipping is possible by setting a value between the pre-correction angle and the post-correction angle calculated from the angle, speed, and PWM cycle.

a. Phase interpolation

<Equation>

```
theta0 = [VExOMEGA] × [VExTPWM] + [VExTHETA]          ; Calculates the value of
                                                        phase interpolation
theta0 = theta0 & 0x0000FFFF
if ([VExMODE]<CLPEN> = 1)                             ; Enables the clipping
  if ([VExOMEGA] ≥ 0)                                 ; On positive rotation
    if ([VExTHETA] ≤ [VExTHTCLP] ≤ theta0)            theta0 = [VExTHTCLP]
    else if (theta0 ≤ [VExTHETA] ≤ [VExTHTCLP])       theta0 = [VExTHTCLP]
    else if ([VExTHTCLP] ≤ theta0 ≤ [VExTHETA])       theta0 = [VExTHTCLP]
  else                                                ; On inverse rotation
    if (theta0 ≤ [VExTHTCLP] ≤ [VExTHETA])            theta0 = [VExTHTCLP]
    else if ([VExTHTCLP] ≤ [VExTHETA] ≤ theta0)       theta0 = [VExTHTCLP]
    else if ([VExTHETA] ≤ theta0 ≤ [VExTHTCLP])       theta0 = [VExTHTCLP]
if ([VExMODE]<PVIEN> = 1)    [VExTHETA] = theta0      ; Updates the [VExTHETA] value when
                                                        phase interpolation is enabled
```

	Register name	Function	Detail
Input	***[VExTHETA]***	Motor phase setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExOMEGA]	Rotational speed setting	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExTPWM]	PWM period rate setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExTHTCLP]	Clipped phase value setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExMODE]	[0] Disables/enables phase interpolation control.	<PVIEN> 0: Disabled 1: Enabled
		[7] Disables/enables phase clipping control when phase interpolation is performed.	<CLPEN> 0: Clipping is disabled 1: Clipping is enabled
Output	***[VExTHETA]***	Motor phase setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)

b. SIN/COS calculation

<Equation>

[VExSINM] = ***[VExSIN]*** ; Stores the previous value (for input process)

[VExCOSM] = ***[VExCOS]***

[VExSIN] = SIN (***[VExTHETA]***) ; Calculates the SIN and COS values

[VExCOS] = SIN (***[VExTHETA]*** + 1 / 4)

if (***[VExFMODE]***<MREGDIS> = 1) ; Confirms that the previous value is not maintained

[VExSINM] = ***[VExSIN]***

[VExCOSM] = ***[VExCOS]***

Note: SIN: Sine calculation

	Register name	Function	Detail
Input	***[VExTHETA]***	Motor phase setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExFMODE]	[9] Enables/disables the use of previous value flags of SIN and COS.	<MREGDIS> 0: Enabled 1: Disabled
	[VExSIN]	Sine value of θ	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExCOS]	Cosine value of θ	
Output	***[VExSIN]***	Sine value of θ	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExCOS]	Cosine value of θ	
	[VExSINM]	Previous sine value	
	[VExCOSM]	Previous cosine value	

3.3.3. Output Voltage Transformation (Coordinate Axis Transformation/Phase Transformation)

Output voltage transformation is performed in two steps:

There are two types of phase transformation tasks, space vector transformation and inverse Clark transformation, and either one is used.

3.3.3.1. Output Coordinate Axis Transformation (Task 7)

In the output coordinate axis task, α-axis and β-axis voltages are calculated based on d-axis voltage, q-axis voltage, sinθ and cosθ.

<Equation>

$$[VExVALPHA] = [VExCOS] \times [VExVD] - [VExSIN] \times [VExVQ] \quad ; \text{Calculates V}\alpha$$
$$[VExVBETA] = [VExSIN] \times [VExVD] + [VExCOS] \times [VExVQ] \quad ; \text{Calculates V}\beta$$

	Register name	Function	Detail
Input	*[VExVD]*	d-axis voltage	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExVQ]	q-axis voltage	
	[VExSIN]	Sine value of θ	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExCOS]	Cosine value of θ	
Output	*[VExVALPHA]*	α-axis voltage	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExVBETA]	β-axis voltage	

3.3.3.2. Output Phase Transformation 1 (Space Vector Transformation) (Task 8)

Output phase transformation 1 determines a sector using α-axis voltage and β-axis voltage. This task calculates the duties of a-phase voltage, b-phase voltage, and c-phase voltage based on space vector transformation in each sector.

This task can be selected either from 2-phase or 3-phase modulation as a modulation type.

a. Sector determination

<Equation>

[VExSECTORM] = *[VExSECTOR]* ; Stores the previous value
valpha = *[VExVALPHA]*
vbeta = *[VExVBETA]*
if (valpha ≥ 0 & vbeta ≥ 0)
  if (|valpha| ≥ |vbeta| × √ (1 / 3))
    if (|valpha| × √ (1 / 3) ≥ |vbeta|) *[VExSECTOR]* = 0
    else *[VExSECTOR]* = 1
  else *[VExSECTOR]* = 2
else if (valpha < 0 & vbeta ≥ 0)
  if (|valpha| < |vbeta| × √ (1 / 3)) *[VExSECTOR]* = 3
  if (|valpha| × √ (1 / 3) < |vbeta|) *[VExSECTOR]* = 4
  else *[VExSECTOR]* = 5
else if (valpha < 0 & vbeta < 0)

```
    if (|valpha| ≥ |vbeta| × √ (1 / 3))
      if (|valpha| × √ (1 / 3) ≥ |vbeta|)     [VExSECTOR] = 6
      else                                    [VExSECTOR] = 7
    else                                      [VExSECTOR] = 8
  else if (valpha ≥ 0 & vbeta < 0)
    if (|valpha| < |vbeta| × √ (1 / 3))       [VExSECTOR] = 9
    else if (|valpha| × √ (1 / 3) < |vbeta|)  [VExSECTOR] = 10
    else                                      [VExSECTOR] = 11
  if ([VExFMODE]<MREGDIS> = 1)                [VExSECTORM] = [VExSECTOR]    ; Confirm that the previous
                                                                              value is disabled
```

	Register name	Function	Detail
Input	***[VExVALPHA]***	α- axis voltage	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExVBETA]	β- axis voltage	
	[VExFMODE]	[9] Enables/disables the use of previous value flags of sector.	<MREGDIS> 0: Enabled 1: Disabled
	[VExSECTOR]	Sector information	4-bit integer data ("0" to "11") * It is processed at the beginning of the equation, it becomes the processing of the previous value.
Output	***[VExSECTOR]***	Sector information	4-bit integer data ("0" to "11")
	[VExSECTORM]	Previous sector information	

b. Space vector transformation
(This example describes only the case where 3-phase modulation is used and ***[VExSECTOR]*** = 0, 1.)

<Equation>

```
  if ([VExVDC] < 1 / 128)      [VExMCTLF]<LVTF> = 1
  else                         [VExMCTLF]<LVTF> = 0
  kc = √3                                                    ; Relative transformation
  if ([VExFMODE]<CCVMD> = 1)   kc = √2                       ; Absolute transformation
  if ([VExSECTOR] = 0, 1)
    t1 = kc /[VExVDC] × (√3 / 2 × [VExVALPHA] -1 / 2 × [VExVBETA])   ; Calculates the t1 period.
    t2 = kc /[VExVDC] × [VExVBETA])                                  ; Calculates the t2 period.
    t3 = 1 -t1 -t2                                           ; Calculates the zero vector period
    if ([VExFMODE]<C2PEN> = 0)                               ; 3-phase modulation
      dutya = t1 + t2 + t3 / 2
      dutyb = t2 + t3 / 2
      dutyc = t3 / 2
    else                                                     ; 2-phase modulation
      dutya = t1 + t2
      dutyb = t2
      dutyc = 0

  [VExVDUTYA] = dutya
  [VExVDUTYB] = dutyb
  [VExVDUTYC] = dutyc
  [VExDUTYZERO] = t3
```

	Register name	Function	Detail
Input	***[VExVALPHA]***	α- axis voltage	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExVBETA]	β- axis voltage	
	[VExVDC]	DC supply voltage	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExSECTOR]	Sector information	4-bit integer data ("0" to "11")
	[VExFMODE]	[0] Selects the modulation mode.	<C2PEN> 0: 3-phase modulation 1: 2-phase modulation
		[13] Selects a phase transformation mode.	<CCVMD> 0: Relative transformation 1: Absolute transformation
Output	***[VExVDUTYA]***	a-phase voltage duty	32-bit fixed-point data (0.0 to 1.0, 31 fractional bits)
	[VExVDUTYB]	b-phase voltage duty	
	[VExVDUTYC]	c-phase voltage duty	
	[VExDUTYZERO]	Zero vector duty	
	[VExMCTLF]	[2] Low-supply voltage flag	<LVTF> 0: ***[VExVDC]*** ≥ 1 / 128 1: ***[VExVDC]*** < 1 / 128

3.3.3.3. Output Phase Transformation 2 (Inverse Clark Transformation) (Task 11)

Output phase transformation 2 determines a sector using α-axis and β-axis voltages. This task calculates the duties of a-phase, b-phase and c-phase voltages based on inverse Clark transformation. This task supports only 3-phase modulation as a modulation type.

In addition, if ***[VExFMODE]***<PHCVDIS> is set to "1", this task can calculate the duty of 2- phase voltage.

a. Sector determination

<Equation>

[VExSECTORM] = ***[VExSECTOR]*** ; Stores the previous value
valpha = ***[VExVALPHA]***
vbeta = ***[VExVBETA]***
if (valpha ≥ 0 & vbeta ≥ 0)
  if (|valpha| ≥ |vbeta| × √ (1 / 3))
    if (|valpha| × √ (1 / 3) ≥ |vbeta|) ***[VExSECTOR]*** = 0
    else ***[VExSECTOR]*** = 1
  else ***[VExSECTOR]*** = 2
else if (valpha < 0 & vbeta ≥ 0)
  if (|valpha| < |vbeta| × √ (1 / 3)) ***[VExSECTOR]*** = 3
  else if (|valpha| × √ (1 / 3) < |vbeta|) ***[VExSECTOR]*** = 4
  else ***[VExSECTOR]*** = 5
else if (valpha < 0 & vbeta < 0)
  if (|valpha| ≥ |vbeta| × √ (1 / 3))
    if (|valpha| × √ (1 / 3) ≥ |vbeta|) ***[VExSECTOR]*** = 6
    else ***[VExSECTOR]*** = 7
  else ***[VExSECTOR]*** = 8
else if (valpha ≥ 0 & vbeta < 0)
  if (|valpha| < |vbeta| × √ (1 / 3)) ***[VExSECTOR]*** = 9
  else if (|valpha| × √ (1 / 3) < |vbeta|) ***[VExSECTOR]*** = 10
  else ***[VExSECTOR]*** = 11

if (***[VExFMODE]***<MREGDIS> = 1) ***[VExSECTORM]*** = ***[VExSECTOR]*** ; Confirm that the previous value is disabled

	Register name	Function	Detail
Input	***[VExVALPHA]***	α-axis voltage	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExVBETA]	β-axis voltage	
	[VExFMODE]	[9] Enables/disables the use of previous value flags of sector.	<MREGDIS> 0: Enabled 1: Disabled
	[VExSECTOR]	Sector information	4-bit integer data ("0" to "11") * It is processed at the beginning of the equation, it becomes the processing of the previous value.
Output	***[VExSECTOR]***	Sector information	4-bit integer data ("0" to "11")
	[VExSECTORM]	Previous sector information	

b. Inverse Clark transformation

<Equation>

if (***[VExVDC]*** < 1 / 128) ***[VExMCTLF]***<LVTF> = 1
else ***[VExMCTLF]***<LVTF> = 0

kc = 1 ; Relative transformation

if (***[VExFMODE]***<CCVMD> = 1) kc = √ (2 / 3) ; Absolute transformation

if (***[VExFMODE]***<PHCVDIS> = 0) ; 3-phase transformation is enabled.

[VExVDUTYA] = kc / ***[VExVDC]*** × ***[VExVALPHA]*** + 1 / 2 ; Va duty

[VExVDUTYB] = kc / ***[VExVDC]*** × (-1 / 2 × ***[VExVALPHA]*** + √3 / 2 × ***[VExVBETA]***) + 1 / 2 ; Vb duty

[VExVDUTYC] = kc / ***[VExVDC]*** × (-1 / 2 × ***[VExVALPHA]*** -√3 / 2 × ***[VExVBETA]***) + 1 / 2 ; Vc duty

else ; Phase transformation is disabled.

[VExVDUTYA] = 1 /***[VExVDC]*** × ***[VExVALPHA]*** + 1 / 2 ; Va duty

[VExVDUTYB] = 1 /***[VExVDC]*** × ***[VExVBETA]*** + 1 / 2 ; Vb duty

***** When phase transformation is disabled, outputs the result of 3-phase transformation to ***[VExVDUTYC]***.

	Register name	Function	Detail
Input	***[VExVALPHA]***	α-axis voltage	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExVBETA]	β-axis voltage	
	[VExVDC]	DC supply voltage	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExFMODE]	[12] Disables/enables phase transformation.	<PHCVDIS> 0: 2-3 phase transformation is enabled. (3-phase AC output) 1: 2-3 phase transformation is disabled. (2-phase AC output)
		[13] Selects a phase transformation mode.	<CCVMD> 0: Relative transformation 1: Absolute transformation
Output	***[VExVDUTYA]***	a-phase voltage duty	32-bit fixed-point data (0.0 to 1.0, 31 fractional bits)
	[VExVDUTYB]	b-phase voltage duty	
	[VExVDUTYC]	c-phase voltage duty	
	[VExMCTLF]	[2] Low-supply voltage flag	<LVTF> 0: ***[VExVDC]*** ≥ 1 / 128 1: ***[VExVDC]*** < 1 / 128

3.3.4. Output Control

Output control unit converts a 3-phase voltage duty into the PMD setting format. The transformation result is set to ***[VExCMPU]***, ***[VExCMPV]***, and ***[VExCMPW]***. According to the output control operation setting, ***[VExOUTCR]*** should be set. Dead time compensation control and PWM output limitation can also be executed.

There are two types of output control task: output control 1 and output control 2. Each task supports different PWM outputs.

Note: When PMD is set to VE mode, the PMD's ***[PMDxCMPU]***/***[PMDxCMPV]***/***[PMDxCMPW]***/***[PMDxMDOUT]*** registers switch to ***[VExCMPU]***/***[VExCMPV]***/***[VExCMPW]***/***[VExOUTCR]***.
Even so, double buffer/triple buffer function of PMD is effective. For details of the buffer function, refer to "Advanced Programmable Motor Control Circuit 2" in the reference manual.
When double buffers are enabled with PMD, while the output control 1 task/output control 2 task/trigger generation task are running, the execution stage is not updated at the execution stage update timing.

Please perform dead time compensation under the condition that current detection is possible.

3.3.4.1. Output Control 1 (Task 0)

Output control 1 task supports normal PWM outputs and PWM outputs in the PWM shift 1. When PWM shift is enabled, if the rotational speed (***[VExOMEGA]***) is lower than the reference of PWM shift switch (***[VExFPWMCHG]***), PWM outputs changes to PWM shift 1.

Note: PWM shift 1 can only be selected in the 1-shunt current detection mode.

a. Output transformation

<Equation>

```
[VExMCTLF]<LAVFM> = [VExMCTLF]<LAVF>          ; Updates a previous value of low-speed flag
[VExMCTLF]<SFT2STM> = [VExMCTLF]<SFT2ST>      ; Updates a previous value of PWM carrier
                                                reverse flag of PWM shift 2
[VExMCTLF] & = 0xB7EE                         ; Current flag is cleared
if (|[VExOMEGA]| ≤ [VExFPWMCHG])
    [VExMCTLF]<LAVF> = 1                      ; Sets the low-speed flag for low-speed
                                                determination
dutya = [VExVDUTYA]
dutyb = [VExVDUTYB]
dutyc = [VExVDUTYC]
if ([VExMCTLF]<LAVF> = 1 & [VExFMODE]<C2PEN> = 1 & [VExFMODE]<SPWMEN> = 1
   & ([VExFMODE]<IDMODE[1]> = 1)              ; PWM shift 1 at low-speed
  if ([VExSECTOR] = 0, 3, 4, 7, 8, 11)        ; Sector determination
    dutya = dutya + [VExDUTYZERO]             ; Zero vector V7 transformation
    dutyb = dutyb + [VExDUTYZERO]
    dutyc = dutyc + [VExDUTYZERO]
pwma = dutya × 0x8000                         ; Transforms the PMD setting value
pwmb = dutyb × 0x8000
pwmc = dutyc × 0x8000
```

	Register name	Function	Detail
Input	***[VExVDUTYA]***	a-phase voltage duty	32-bit fixed-point data (0.0 to 1.0, 31 fractional bits)
	[VExVDUTYB]	b-phase voltage duty	
	[VExVDUTYC]	c-phase voltage duty	
	[VExSECTOR]	Sector information	4-bit integer data ("0" to "11")
	[VExOMEGA]	Rotational speed setting	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExFPWMCHG]	PWM switching speed setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits) Shift switch is disabled when ***[VExFPWMCHG]*** = 0x0000
	[VExFMODE]	[0] Selects the modulation mode.	<C2PEN> 0: 3-phase modulation 1: 2-phase modulation
		[1] Disables/enables PWM shift.	<SPWMEN> 0: Disabled 1: Enabled
		[3:2] Selects current detection mode.	<IDMODE[1:0]> 00: 3-shunt 01: 2 sensors 10, 11: 1-shunt
	[VExDUTYZERO]	Zero vector duty	32-bit fixed-point data (0.0 to 1.0, 31 fractional bits)
	[VExMCTLF]	[1] Previous value of the <LAVF>	<LAVFM>
		[14] PWM carrier reverse flag of PWM shift 2	<SFT2ST>
Output	***[VExMCTLF]***	[1:0] Low-speed flag	<LAVFM>, <LAVF>
		[11] Indicates the excess flag for PWM output limitation	<PWMOVF>
		[15:14] PWM carrier reverse flag of PWM shift 2	<SFT2STM>, <SFT2ST>

Note: ***[VExTMPREG0]*** is updated to an undefined value because it is used to temporarily save internal data during task execution.

b. PWM output limitation

<Equation>

```
if (VExPWMMAX] = 0)      max = 0x8000
else                     max = VExPWMMAX]
if ((pwma > max) & (0x8000 > max))                    ; Checks the PWM upper limit for U-phase.
  if ((VExMODE]<PWMFLEN> = 0) | (pwma < 0x8000))      ; Confirms the duty of 100% of the output
                                                        limitation.
    pwma = max
    VExMCTLF]<PWMOVF> = 1
min = VExPWMMIN]
if ((pwma < min) & (min > 0))                         ; Checks the PWM lower limit for U-phase.
  if ((VExMODE]<PWMBLEN> = 0) | (pwma > 0))           ; Confirms the duty of 0% of the output
                                                        limitation.
    pwma = min
    VExMCTLF]<PWMOVF> = 1
```

(Calculates the other 2-phase in the same manner)

	Register name	Function	Detail
Input	***[VExPWMMAX]***	PWM upper limit setting	16-bit data ("0x0000" to "0x8000")
	[VExPWMMIN]	PWM lower limit setting	
	[VExMODE]	[12] Disables/enables a duty of 0% setting when the lower limit is set.	<PWMBLEN> 0: Duty of 0% setting is disabled. 1: Duty of 0% setting is enabled.
		[13] Disables/enables a duty of 100% setting when the upper limit is set.	<PWMFLEN> 0: Duty of 100% setting is disabled. 1: Duty of 100% setting is enabled.
Output	***[VExMCTLF]***	[11] Indicates the excess flag of the PWM output limitation.	<PWMOVF>

Note: PWM output limitation is disabled when PWM shift 1 ***[VExFMODE]*** = 1 (shift enabled) and ***[VExMCTLF]*** = 1 (low speed).

c. Dead time compensation

<Equation>

```
if (0 < pwma < 0x8000)                    dt = [VExDTC]
else                                      dt = 0
if ([VExDTCS]<IASTS[2:0]> = 001, 101)                          ; On Positive current
  if ([VExMODE]<PMDDTCEN> = 1)                                 ; Sets the dead time correction of the PMD
    if (pwma > (0x8000 - 2 × dt))         pwma = (0x8000 + pwma) / 2
  else                                    pwma = pwma + dt
  if (([VExMODE]<PWMFLEN> = 1) & (max < 0x8000))               ; Confirms the duty of 100%
                                                                 of the output limitation
    if (pwma > (0x8000-1))                pwma = 0x8000 -1     ; Output limitation after correction.
  else
    if (pwma > 0x8000)                    pwma = 0x8000        ; Output limitation after correction.
else if ([VExDTCS]<IASTS[2:0]> = 011, 111)                     ; On negative current
  if ([VExMODE]<PMDDTCEN> = 1)                                 ; Sets the dead time correction of the PMD
    if (pwma < (2 × dt))                  pwma = pwma / 2
  else                                    pwma = pwma - dt
  if (([VExMODE]<PWMBLEN> = 1) & (min > 0))                    ; Confirms the duty of 0%
                                                                 of the output limitation.
    if (pwma < 1)                         pwma = 1             ; Output limitation after correction.
  else
    if (pwma < 0)                         pwma = 0             ; Output limitation after correction.
```

(Calculates the other 2-phase in the same manner.)

	Register name	Function	Detail
Input	***[VExDTC]***	Dead time compensation	16-bit data ("0x0000" to "0x1FFF")
	[VExMODE]	[12] Disables/enables a duty of 0% setting when the lower limit is set.	<PWMBLEN> 0: Duty of 0% setting is disabled. 1: Duty of 0% setting is enabled.
		[13] Disables/enables a duty of 100% setting when the upper limit is set.	<PWMFLEN> 0: Duty of 100% setting is disabled. 1: Duty of 100% setting is enabled.
		[14] Setting that corresponds to the dead time correction control of the PMD when the dead time compensation is performed.	<PMDDTCEN> 0: Dead time correction of the PMD is disabled. 1: Dead time correction of the PMD is enabled.
	[VExDTCS]	Dead time compensation control/status	<ICSTS[2:0]>, <IBSTS[2:0]>, <IASTS[2:0]> 000, 010, 100, 110: Polarity determination is undefined. Dead time compensation is not performed. 001, 101: Positive current is determined. Adds the ***[VExDTC]*** value when dead time compensation is performed. 011, 111: Negative current is determined. Subtracts the ***[VExDTC]*** value when dead time compensation is performed.

d. Output control/PWM shift 1 transformation

<Equation>

```
outcr = 0x1FF                                                    ; All phases center on
if (***[VExMCTLF]***<LAVF> = 1 & ***[VExFMODE]***<C2PEN> = 1
& ***[VExFMODE]***<SPWMEN> = 1 & ***[VExFMODE]***<IDMODE[1]> = 1)    ; PWM shift 1 at low-speed
   if (***[VExSECTOR]*** = 0, 1, 2, 11)                          ; V-phase center off
         outcr = 0x1F3                                           ; Reverses a V-phase carrier
   else if (***[VExSECTOR]*** = 3, 4, 5, 6)                      ; W-phase center off
         outcr = 0x1CF                                           ; Reverses a W-phase carrier
   else if (***[VExSECTOR]*** = 7, 8, 9, 10)                     ; U-phase center off
         outcr = 0x1FC                                           ; Reverses a U-phase carrier
if (***[VExMODE]***<OCRMD[1:0]> = 00, 11)                        ; Output off
         outcr = 0x000
else if (***[VExMODE]***<OCRMD[1:0]> = 10)                       ; Short circuit brake
         outcr = 0x015
***[VExCMPU]*** = pwma
***[VExCMPV]*** = pwmb
***[VExCMPW]*** = pwmc
***[VExOUTCR]*** = outcr

[Pulse width difference check. All off-width check]
dif1 = | pwmc - pwma |
dif2 = | pwmb - pwmc |
dif3 = | pwma - pwmb |
difmin = dif1
if (dif2 < difmin)    difmin = dif2
```

```
if (dif3 < difmin)    difmin = dif3                          ; Minimum width difference
difmax = (dif1 + dif2 + dif3) / 2                            ; Maximum width difference

difmid = difmax – difmin                                     ; Medium width difference

dmax = pwma
if (pwmb > dmax)   dmax = pwmb
if (pwmc > dmax)   dmax = pwmc                               ; Maximum width
offmin = 1-dmax                                              ; Minimum off-width

dmin = pwma
if (pwmb < dmin) dmin = pwmb
if (pwmc < dmin) dmin = pwmc                                 ; Minimum width

dmid = pwma
if (dmid = dmax) dmid = pwmb
if (dmid = dmin) dmid = pwmc
if (dmid = dmax) dmid = pwmb                                 ; Medium width
offmid = 1 – dmid                                            ; Medium off-width

if (/VExMCTLF/<LAVF> = 1 & /VExFMODE/<C2PEN> = 1 & /VExFMODE/<SPWMEN> = 1
   & (/VExFMODE/<IDMODE[1]> = 1)                             ; PWM shift 1 at low-speed
if (/VExSECTOR/ = 0, 3, 4, 7, 8, 11)
    if (dmin > offmid)        difmin = offmid                ; Medium off-width
    else                      difmin = dmin                  ; Minimum width
  else
    if (dmid > offmin)        difmin = offmin                ; Minimum off-width
    else                      difmin = dmid                  ; Medium width
  difmid = difmin
else if (/VExFMODE/<C2PEN> = 1)                              ; Normal PWM/2-phase modulation
  if (/VExSECTOR/ = 0, 3, 4, 7, 8, 11) difmin = difmin × 2   ; Minimum width difference =
                                                             correction for minimum width sector
  if (difmid < difmin) difmin = difmid                       ; Check minimum width difference
                                                               after correction
```

	Register name	Function	Detail
Input	***[VExSECTOR]***	Sector information	4-bit integer data ("0" to "11")
	[VExMODE]	[3:2] Specifies output control.	<OCRMD[1:0]> 00: Output is disabled. 01: Output is enabled. 10: Short circuit brake 11: reserved
	[VExFMODE]	[0] Selects the modulation mode.	<C2PEN> 0: 3-phase modulation 1: 2-phase modulation
		[1] Disables/enables PWM shift	<SPWMEN> 0: Disabled 1: Enabled
	[VExMCTLF]	[0] Low-speed flag	<LAVF>
Output	***[VExCMPU]***	U-phase PWM duty	16-bit data ("0x0000" to "0x8000")
	[VExCMPV]	V-phase PWM duty	
	[VExCMPW]	W-phase PWM duty	
	[VExOUTCR]	PMD output control	9-bit setting

3.3.4.2. Output Control 2 (Task 9)

Output control 2 task corresponds to PWM output of PWM shift 2.
When the PWM output in PWM shift 2 is set, enable PWM shift (***[VExFMODE]***<SPWMEN> = 1) and select the value other than "00" for PWM shift mode selection (***[VExFMODE]***<SPWMMD[1:0]>).

Note: PWM shift can be selected only in 1-shunt current detection mode.

a. Output transformation

<Equation>

[VExMCTLF]<LAVFM> = ***[VExMCTLF]***<LAVF> ; Updates a previous value of low-speed flag
[VExMCTLF]<STF2STM> = ***[VExMCTLF]***<SFT2ST> ; Updates a previous value of PWM carrier reverse flag of PWM shift 2
[VExMCTLF] & = 0xB7EE ; Current flag is cleared
if (***[VExOMEGA]*** < ***[VExFPWMCHG]***) ***[VExMCTLF]*** <LAVF> = 1
; Sets the low-speed flag for low-speed determination

if (***[VExFMODE]***<SPWMEN> = 1 & ***[VExFMODE]***<IDMODE[1]> = 1
& ***[VExFMODE]***<SPWMMD[1:0]> ≠00)
pwmofs = ***[VExPWMOFS]***
else pwmofs = 0

pwma = ***[VExVDUTYA]*** × 0x8000 ; Transforms the PMD setting value
pwmb = ***[VExVDUTYB]*** × 0x8000
pwmc = ***[VExVDUTYC]*** × 0x8000

	Register name	Function	Detail
Input	***[VExVDUTYA]***	a-phase voltage duty	32-bit fixed-point data (0.0 to 1.0, 31 fractional bits)
	[VExVDUTYB]	b-phase voltage duty	
	[VExVDUTYC]	c-phase voltage duty	
	[VExOMEGA]	Rotational speed setting	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExFPWMCHG]	PWM switching speed setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits) Shift switch is disabled when ***[VExFPWMCHG]*** = 0x0000
	[VExFMODE]	[1] Disables/enables PWM shift.	<SPWMEN> 0: Disabled 1: Enabled
		[3:2] Selects current detection mode.	<IDMODE[1:0]> 00: 3-shunt 01: 2 sensors 10, 11: 1-shunt
		[15:14] Selects a PWM shift mode.	<SPWMMD[1:0]> 00: Shift 1 01: Shift 2 (U-phase center) 10: Shift 2 (V-phase center) 11: Shift 2 (W-phase center)
	[VExPWMOFS]	PWM shift 2 offset	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExMCTLF]	[0] Low-speed flag	<LAVF>
		[14] A PWM carrier reverse flag of PWM shift 2	<SFT2ST>
Output	***[VExMCTLF]***	[1:0] Low-speed flag	<LAVFM>, <LAVF>
		[11] Indicates the excess flag for PWM output limitation	<PWMOVF>
		[15:14] A PWM carrier reverse flag of PWM shift 2	<SFT2STM>, <SFT2ST>

Note: ***[VExTMPREG0]*** is updated to an undefined value because it is used for temporary backup of internal data during task execution.

b. PWM output limitation

<Equation>

```
if (VExPWMMAX] = 0)      max = 0x8000
else                     max = [VExPWMMAX]
if ((pwma > max) & (0x8000 > max))                    ; Checks the PWM upper limit for U-phase.
  if ((VExMODE]<PWMFLEN> = 0) | (pwma < 0x8000)); Confirms the duty of 100% of the output
                                                        limitation.
    pwma = max
    VExMCTLF]<PWMOVF> = 1
min = VExPWMMIN]
if ((pwma < min) & (min > 0))                         ; Checks the PWM lower limit for U-phase.
  if ((VExMODE]<PWMBLEN> = 0) | (pwma > 0))           ; Confirms the duty of 0% of the output
                                                        limitation.
    pwma = min
    VExMCTLF]<PWMOVF> = 1
```

(Calculates the other 2-phase in the same manner)

	Register name	Function	Detail
Input	***[VExPWMMAX]***	PWM upper limit setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExPWMMIN]	PWM lower limit setting	
	[VExMODE]	[12] Disables/enables a duty of 0% setting when the lower limit is set.	<PWMBLEN> 0: Duty of 0% setting is disabled. 1: Duty of 0% setting is enabled.
		[13] Disables/enables a duty of 100% setting when the upper limit is set.	<PWMFLEN> 0: Duty of 100% setting is disabled. 1: Duty of 100% setting is enabled.
Output	***[VExMCTLF]***	[11] Indicates the excess flag of the PWM output limitation.	<PWMOVF>

c. Dead time compensation

<Equation>

```
if (VExMODE<PMDDTCEN> = 0) dt = 0                    ; When correction is prohibited, set the
                                                       correction amount to 0.
else                                  dt = VExDTC
if (0 < pwma < 0x8000)                dt = VExDTC
else                                  dt = 0
if (VExDTCS<IASTS[2:0]> = 001, 101)                  ; On positive current
  if (VExMODE<PMDDTCEN> = 1)                         ; Sets the dead time correction of the PMD
    if (pwma > (0x8000 - 2 × dt))pwma = (0x8000 + pwma) / 2
  else                                pwma = pwma + dt
  if ((VExMODE<PWMFLEN> = 1) & (max < 0x8000))       ; Confirms the duty of 100%
                                                       of the output limitation
    if (pwma > (0x8000 - 1))          pwma = 0x8000 - 1  ; Output limitation after correction.
  else
    if (pwma > 0x8000)                pwma = 0x8000      ; Output limitation after correction.
else if (VExDTCS<IASTS[2:0]> = 011, 111)             ; On negative current
  if (VExMODE<PMDDTCEN> = 1)                         ; Sets the dead time correction of the PMD
    if (pwma < (2 × dt))              pwma = pwma / 2
  else                                pwma = pwma - dt
  if ((VExMODE<PWMBLEN> = 1) & (min > 0))            ; Confirms the duty of 0%
                                                       of the output limitation.
    if (pwma < 1)                     pwma = 1           ; Output limitation after correction.
  else
    if (pwma < 0)                     pwma = 0           ; Output limitation after correction.

    (Calculates the other 2-phase in the same manner.)
```

	Register name	Function	Detail
Input	***[VExDTC]***	Dead time compensation	16-bit data ("0x0000" to "0x1FFF")
	[VExMODE]	[12] Disables/enables a duty of 0% setting when the lower limit is set.	<PWMBLEN> 0: Duty of 0% setting is disabled. 1: Duty of 0% setting is enabled.
		[13] Disables/enables a duty of 100% setting when the upper limit is set.	<PWMFLEN> 0: Duty of 100% setting is disabled. 1: Duty of 100% setting is enabled.
		[14] Setting that corresponds to the dead time correction control of the PMD when the dead time compensation is performed.	<PMDDTCEN> 0: Dead time correction of the PMD is disabled. 1: Dead time correction of the PMD is enabled.
	[VExDTCS]	Dead time compensation control/status	<ICSTS[2:0]>, <IBSTS[2:0]>, <IASTS[2:0]> 000, 010, 100, 110: Polarity determination is undefined. Dead time compensation is not performed. 001, 101: Positive current is determined. Adds the ***[VExDTC]*** value when dead time compensation is performed. 011, 111: Negative current is determined. Subtracts the ***[VExDTC]*** value when dead time compensation is performed.

d. Output control /PWM shift 2 transformation

<Equation>

```
outcr = 0x1FF                                                    ; All phases center on
if ((VExFMODE]<IDMODE[1]> = 1) & (VExFMODE]<SPWMEN> = 1))   ; Shift 2 is enabled in
                                                                   1-shunt
 if (VExFPWMCHG] = 0 | VExMCTLF]<LAVF> = 1)
   outcr = 0x1FF                                                 ; No carrier reverse
 else                                                            ; Shift 2 control area determination

   if (VExFMODE]<SPWMMD[1:0]> = 01)                              ; Shift 2, U-phase reference
     if (pwma < 0x4000)                  outcr = 0x1FC           ; U-phase carrier reverse control
     else                                outcr = 0x1FF           ; No carrier reverse
   else if (VExFMODE]<SPWMMD[1:0]> = 10)                         ; Shift 2, V-phase reference
     if (pwmb < 0x4000)                  outcr = 0x1F3           ; V-phase carrier reverse control
     else                                outcr = 0x1FF           ; No carrier reverse
   else if (VExFMODE]<SPWMMD[1:0]> = 11)                         ; Shift 2, W-phase reference
     if (pwmc < 0x4000)                  outcr = 0x1CF           ; W-phase carrier reverse control
     else                                outcr = 0x1FF           ; No carrier reverse

if (VExMODE]<OCRMD[1:0]> = 00, 11)                 outcr = 0x000     ; Output off
else if (VExMODE]<OCRMD[1:0]> = 10)                outcr = 0x015     ; Short circuit brake
VExCMPU] = pwma
VExCMPV] = pwmb
VExCMPW] = pwmc
VExOUTCR] = outcr
```

	Register name	Function	Detail
Input	***[VExFPWMCHG]***	PWM switching speed setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits) Shift switch is disabled when ***[VExFPWMCHG]*** = 0x0000
	[VExMCTLF]	[0] Low-speed flag	<LAVF>
	[VExMODE]	[3:2] Specifies output control	<OCRMD[1:0]> 00: Output is disabled. 01: Output is enabled. 10: Short circuit brake. 11: Reserved
	[VExFMODE]	[1] Disables/enables PWM shift.	<SPWMEN> 0: Disabled 1: Enabled
		[3:2] Selects current detection mode.	<IDMODE[1:0]> 00: 3-shunt 01: 2 sensors 10, 11: 1-shunt
		[15:14] Selects a PWM shift mode.	<SPWMMD[1:0]> 00: Shift 1 01: Shift 2 (U-phase center) 10: Shift 2 (V-phase center) 11: Shift 2 (W-phase center)
Output	***[VExCMPU]***	U-phase PWM duty	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExCMPV]	V-phase PWM duty	
	[VExCMPW]	W-phase PWM duty	
	[VExOUTCR]	PMD output control	9-bit setting
	[VExMCTLF]	[14] A PWM carrier reverse flag of PWM shift 2	<SFT2ST> If (reference phase duty < 50%) <SFT2ST> = 1, with reverse carrier of reference phase.

3.3.5. Trigger Generation (Task 1)

The trigger generation unit calculates the synchronous trigger timing for ADC from the PWM duty setting values (***[VExCMPU]***, ***[VExCMPV]***, and ***[VExCMPW]***) based on the current detection method, and sets it to the ***[VExTRGCMP0]*** and ***[VExTRGCMP1]*** registers.

Note 1: ***[VExTRGCMP0]*** and ***[VExTRGCMP1]*** are updated only in 1-shunt current detection

Note 2: ***[VExTRGCMP0]*** and ***[VExTRGCMP1]*** are not updated when PWM shift 2 is selected.

Note 3: Please select "Compare to basic carrier" in the trigger control register comparison carrier selection ***[PMDxTRGCR]***<CARSEL> of the PMD.

Note 4: ***[VExTRGCMP0]*** / ***[VExTRGCMP1]*** / ***[VExTRGSEL]*** switches to ***[PMDxTRGCMP0]*** / ***[PMDxTRGCMP1]*** / ***[PMDxTRGSEL]*** register of PMD when PMD is set to VE mode. Even in that case, double buffer / triple buffer function of PMD is effective. When double buffers / triple buffers are enabled in the PMD, the execution stage is updated even when the update timing of the double buffer / triple buffer execution stage is reached while the output control 1 task/output control 2 task/trigger generation task is being executed not.

Note 5: For details of PMD, refer to "Advanced Programmable Motor Control Circuit 2" in the reference manual.

Current detection mode <IDMODE[1:0]>	PWM shift setting <SPWMEN> <SPWMMD>	Modulation mode <C2PEN>	Low-speed flag <LAVF>	Trigger generation formula
3-shunt	-	2-phase/ 3-phase	-	(No trigger generation)
2 sensors	-	2-phase/ 3-phase	-	(No trigger generation)
1-shunt (Latter half)	Disable PWM shift	3-phase	-	***[VExTRGCMP0]*** = 0x3FFF + (dmin + dmid) / 4 + ***[VExTRGCRC]*** ***[VExTRGCMP1]*** = 0x3FFF + (dmax + dmid) / 4 + ***[VExTRGCRC]***
		2-phase	-	***[VExTRGCMP0]*** = 0x3FFF + ***[VExTRGCRC]*** ***[VExTRGCMP1]*** = 0x3FFF + (dmax + dmid) / 4 + ***[VExTRGCRC]***
	PWM shift 1	2-phase	High-speed	***[VExTRGCMP0]*** = 0x3FFF + ***[VExTRGCRC]*** ***[VExTRGCMP1]*** = 0x3FFF + (dmax + dmid) / 4 + ***[VExTRGCRC]***
			Low-speed	***[VExTRGCMP0]*** = 0x3FFF + ***[VExTRGCRC]*** ***[VExTRGCMP1]*** = 0x7FFF - ***[VExTADC]*** + ***[VExTRGCRC]***
	PWM shift 2	3-phase	-	(No trigger generation)
1-shunt (First half)	Disable PWM shift	3-phase	-	***[VExTRGCMP0]*** = 0x3FFF - (dmax + dmid) / 4 + ***[VExTRGCRC]*** ***[VExTRGCMP1]*** = 0x3FFF - (dmin + dmid) / 4 + ***[VExTRGCRC]***
		2-phase	-	***[VExTRGCMP0]*** = 0x3FFF - (dmax + dmid) / 4 + ***[VExTRGCRC]*** ***[VExTRGCMP1]*** = 0x3FFF + ***[VExTRGCRC]***
	PWM shift 1	2-phase	High-speed	***[VExTRGCMP0]*** = 0x3FFF - (dmax + dmid) / 4 + ***[VExTRGCRC]*** ***[VExTRGCMP1]*** = 0x3FFF + ***[VExTRGCRC]***
			Low-speed	***[VExTRGCMP0]*** = 0x0000 + ***[VExTADC]*** + ***[VExTRGCRC]*** ***[VExTRGCMP1]*** = 0x3FFF + ***[VExTRGCRC]***
	PWM shift 2	3-phase	-	(No trigger generation)

Note: Values of dmin: minimum duty phase, dmax: maximum duty phase, and dmid: intermediate duty phase are determined by ***[VExCMPU]***, ***[VExCMPV]***, and ***[VExCMPW]***. For details, please refer to Output Control 1 (Task 0).

Note: When ***[VExMODE]***<OCRMD[1:0]> ≠ 01, ***[VExTRGCMP0]*** = 0x3FFF + ***[VExTRGCRC]***, ***[VExTRGCMP1]*** = 0x7FFF - ***[VExTRGCRC]***.

Note: When ***[VExCODE]***<CRECEN> = 0, ***[VExTRGCRC]*** is not added.

Note: However, this process is unnecessary for the one-shunt current detection system, and it is recommend that ***[VExMODE]*** is "0".

	Register name	Function	Detail
Input	***[VExCMPU]***	U-phase PWM duty	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExCMPV]	V-phase PWM duty	
	[VExCMPW]	W-phase PWM duty	
	[VExTADC]	ADC conversion time setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExTRGCRC]	Synchronous trigger correction setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExSECTOR]	Sector information	4-bit integer data ("0" to "11")
	[VExSECTORM]	Previous sector information	4-bit integer data ("0" to "11")
	[VExMODE]	[1] Specifies zero-current detection control.	<ZIEN> 0: Normal current detection 1: Zero-current detection
		[3:2] Specifies output control.	<OCRMD[1:0]> 00: Output is disabled. 01: Output is enabled. 10: Short circuit brake. 11: Reserved
	[VExFMODE]	[0] Selects the modulation mode.	<C2PEN> 0: 3-phase modulation 1: 2-phase modulation
		[1] Disables/enables PWM shift.	<SPWMEN> 0: Disabled 1: Enabled
		[3:2] Selects current detection mode.	<IDMODE[1:0]> 00: 3-shunt 01: 2 sensors 10, 11: 1-shunt
		[8] Disables/enables trigger correction	<CRCEN> 0: Disabled 1: Enabled
		[15:14] Selects a PWM shift mode.	<SPWMMD[1:0]> 00: Shift 1 01: Shift 2 (U-phase center) 10: Shift 2 (V-phase center) 11: Shift 2 (W-phase center)
	[VExMCTLF]	[0] Low-speed flag	<LAVF>
Output	***[VExTRGCMP0]***	PMD trigger 0 timing setting	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExTRGCMP1]	PMD trigger 1 timing setting	
	[VExSECTORM]	Previous sector information	4-bit integer data ("0" to "11")

3.3.6. Input Process

In the input process, the vector engine reads conversion results and phase information from the ADC. Depending on the current detection method and PWM shift mode setting, the vector engine converts the phase current data and the voltage data into the fixed-point format and stores them in predefined registers.

In zero-current detection mode, current detection results are stored in the zero-current register.

The vector engine specifies the hysteresis width to determine the current polarity for dead time compensation control.

There are two types of input process: Input process 1 task and input process 2 task. Each task supports different current detection method.

3.3.6.1. Input Process 1 (Task 2)

Input process 1 task supports 3-shunt current detection (only 2-phase current detection (Note 1)) and 1-shunt current detection. However, 1-shunt current detection is not available in PWM shift 2 (Note 2).

Note 1: Current detection result is used only for two phases. Remained one phase is calculated.

Note 2: PWM shift can only be selected in 1-shunt current detection mode.

a. Input transformation

<Equation>

[VDC fixed-point transformation/store]

if (***[VExMODE]***<VDCSEL> = 0) ***[VExVDC]*** = [DC voltage] / 2
else ***[VExVDCL]*** = [DC voltage] / 2
lavfm = ***[VExMCTLF]***<LAVFM> ; Previous low-speed flag
if (***[VExFMODE]***<MREGDIS> = 1) ; Previous value invalid.
lavfm = ***[VExMCTLF]***<LAVF> ; Current low-speed flag
if (***[VExFMODE]***<IDMODE[1]> = 0 | ***[VExFMODE]***<SPWMEN> = 0 | ***[VExFMODE]***<C2PEN> = 0)
lavfm = 0

[Current 1 read] *: When zero current detect, change ***[VExIAADC]→[VExIAO],[VExIBADC]→[VExIBO],[VExICADC] →[VExICO]***.

if (***[VExFMODE]***<IDMODE[1:0]> = 10, 11) ; 1-shunt
if (lavfm = 0) ; Normal PWM
if (***[VExSECTORM]*** = 4, 5, 6, 7) ***[VExIAADC]*** = [Current 1]
else if (***[VExSECTORM]*** = 8, 9, 10, 11) ***[VExIBADC]*** = [Current 1]
else if (***[VExSECTORM]*** = 0, 1, 2, 3) ***[VExICADC]*** = [Current 1]
else if (lavfm = 1) ; PWM shift 1
if (***[VExSECTORM]*** = 1, 2, 7, 8) ***[VExIAADC]*** = [Current 1]
else if (***[VExSECTORM]*** = 0, 5, 6, 11) ***[VExIBADC]*** = [Current 1]
else if (***[VExSECTORM]*** = 3, 4, 9, 10) ***[VExICADC]*** = [Current 1]
if (***[VExFMODE]***<IDMODE[1:0]> = 00, 01) ; 3-shunt, 2 sensors
if ([Current 1 phase information] = 1) ***[VExIAADC]*** = [Current 1]
else if ([Current 1 phase information] = 2) ***[VExIBADC]*** = [Current 1]
else if ([Current 1 phase information] = 3) ***[VExICADC]*** = [Current 1]

[Current 2 read] *: When zero current detect, change ***[VExIAADC]→[VExIAO],[VExIBADC]→[VExIBO],[VExICADC] →[VExICO]***.

```
if (VExFMODE<IDMODE[1:0]> = 10, 11)                    ; 1-shunt
  if (lavfm = 0)                                       ; Normal PWM
    if (VExSECTORM = 0, 1, 10, 11)        VExIAADC = [Current 2]
    else if (VExSECTORM = 2, 3, 4, 5)     VExIBADC = [Current 2]
    else if (VExSECTORM = 6, 7, 8, 9)     VExICADC = [Current 2]
  else if (lavfm = 1)                                  ; PWM shift 1
    if (VExSECTORM = 3, 4, 9, 10)         VExIAADC = [Current 2]
    else if (VExSECTORM = 1, 2, 7, 8)     VExIBADC = [Current 2]
    else if (VExSECTORM = 0, 5, 6, 11)    VExICADC = [Current 2]
if (VExFMODE<IDMODE[1:0]> = 00, 01)                    ; 3-shunt, 2 sensors
    if ([Current 2 phase information] = 1)    VExIAADC = [Current 2]
    else if ([Current 2 phase information] = 2)  VExIBADC = [Current 2]
    else if ([Current 2 phase information] = 3)  VExICADC = [Current 2]
```

[Current 3 read] *: When zero current detect, change ***[VExIAADC]→[VExIAO],[VExIBADC]→[VExIBO],[VExICADC] →[VExICO]***.

```
if (VExFMODE <IDMODE[1:0]> = 00, 01)                   ; Except 1-shunt
    if ([Current 3 phase information] = 1)    VExIAADC = [Current 3]
    else if ([Current 3 phase information] = 2)  VExIBADC = [Current 3]
    else if ([Current 3 phase information] = 3)  VExICADC = [Current 3]
```

[Current fixed-point transformation]

```
ia = VExIAO - VExIAADC
ib = VExIBO - VExIBADC
ic = VExICO - VExICADC
if (VExFMODE<IDMODE[1:0]> = 10, 11)                    ; 1-shunt
  if (lavfm = 0)                                       ; Normal PWM
    if (VExSECTORM = 0, 1, 10, 11)        ia = -ia
    else if (VExSECTORM = 2, 3, 4, 5)     ib = -ib
    else if (VExSECTORM = 6, 7, 8, 9)     ic = -ic
  else if (lavfm = 1)                                  ; PWM shift 1
    if (VExSECTORM = 1, 2, 5, 6, 9, 10)   ia = -ia
                                          ib = -ib
                                          ic = -ic
```

[Current 3 calculation]

```
n = 6 - [Current 2 phase information] -[Current 1 phase information]  ; Calculates the phase number
                                                                        of current 3
if (n = 1)          ia = -ib -ic
else if (n = 2)     ib = -ic -ia
else if (n = 3)     ic = -ia -ib
```

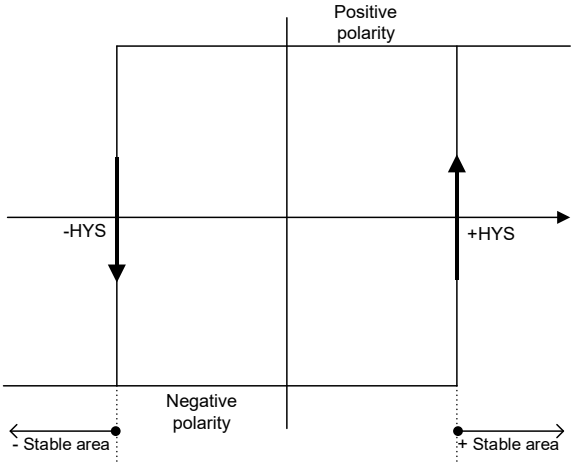
[Current storage]

```
VExIA = ia
VExIB = ib
VExIC = ic
```

	Register name/ signal name	Function	Detail
Input	ADxREG0	Input of the ADC conversion result	16-bit data (The conversion result is stored in the upper 12 bits.) ADxREG0: Current 1 input ADxREG1: Current 2 input ADxREG2: Current 3 input ADxREG3: DC voltage input
	ADxREG1		
	ADxREG2		
	ADxREG3		
	[VExSECTORM]	Previous sector information	4-bit integer data ("0" to "11")
	[VExMODE]	[1] Specifies zero-current detection control.	<ZIEN> 0: Normal current detection 1: Zero-current detection
		[4] Selects the supply voltage store.	<VDCSEL> 0: Stored in ***[VExVDC]*** 1: Stored in ***[VExVDCL]***
	[VExFMODE]	[0] Selects the modulation mode.	<CP2EN> 0: 3-phase modulation 1: 2-phase modulation
		[1] Disables/enables PWM shift.	<SPWMEN> 0: Disabled 1: Enabled
		[3:2] Selects current detection mode.	<IDMODE[1:0]> 00: 3-shunt 01: 2 sensors 10, 11: 1-shunt
		[9] Enables/disables the use of previous value flags of ***[VExMCTLF]***.	<MREGDIS> 0: Enabled 1: Disabled
	[VExMCTLF]	[1:0] Low-speed flag	<LAVFM>, <LAVF>
	[VExIAO]	a-phase zero-current	16-bit data (The result is stored in the upper 12 bits.)
	[VExIBO]	b-phase zero-current	
	[VExICO]	c-phase zero-current	
Output	***[VExVDC]***	DC supply voltage	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExVDCL]	DC2 supply voltage	
	[VExIAADC]	ADC conversion result for a-phase current	16-bit data (The result is stored in the upper 12 bits.)
	[VExIBADC]	ADC conversion result for b-phase current	
	[VExICADC]	ADC conversion result for c-phase current	
	[VExIA]	a-phase current	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExIB]	b-phase current	
	[VExIC]	c-phase current	
	[VExIAO]	a-phase zero-current	16-bit data (The result is stored in the upper 12 bits.)
	[VExIBO]	b-phase zero-current	
	[VExICO]	c-phase zero-current	

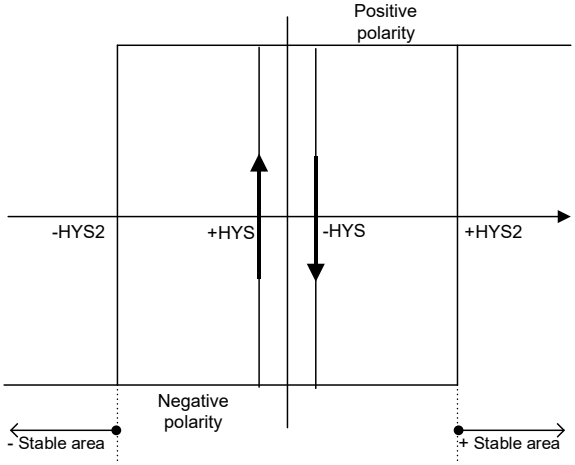
b. Current polarity determination

【**Hysteresis**: 0 ≥ HYS】



[I > HYS, I < -HYS]

【**Reverse hysteresis**: HYS < 0】



[|HYS2| ≤ I, I ≤ -|HYS2|, I > HYS, I < -HYS]

	Register name	Function	Detail
Input	***[VExHYS]***	Hysteresis width for current polarity determination	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExHYS2]	Hysteresis width 2 for current polarity determination	Polarity determination level on reverse hysteresis (***[VExHYS]*** < 0). 16-bit fixed-point data (0 to 1.0, 15 fractional bits) This register is invalid when ***[VExHYS]*** ≥ 0.
	[VExDTCS]	Dead time compensation control/status	<ICSTS[2:0]>, <IBSTS[2:0]>, <IASTS[2:0]> 000, 010, 100, 110: Polarity determination is undefined. Dead time compensation is not performed. 001, 101: Positive current is determined. Adds the ***[VExDTC]*** value when dead time compensation is performed. 011, 111: Negative current is determined. Subtracts the ***[VExDTC]*** value when dead time compensation is performed.
	[VExMODE]	[15] Disables/enables current polarity determination.	<IPDEN> 0: Disabled 1: Enabled.
	[VExFMODE]	[9] Enables/disables the use of previous value flags of ***[VExMCTLF]***.	<MREGDIS> 0: Enabled 1: Disabled
	[VExIA]	a-phase current	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExIB]	b-phase current	
	[VExIC]	c-phase current	
Output	***[VExDTCS]***	Dead time compensation control/status	<ICSTS[2:0]>, <IBSTS[2:0]>, <IASTS[2:0]> 000, 010, 100, 110: Polarity determination is undefined. Dead time compensation is not performed. 001, 101: Positive current is determined. Adds the ***[VExDTC]*** value when dead time compensation is performed. 011, 111: Negative current is determined. Subtracts the ***[VExDTC]*** value when dead time compensation is performed.

3.3.6.2. Input Process 2 (Task 10)

Input process 2 task supports 2-sensor current detection. It also supports 1-shunt current detection when PWM outputting in PWM shift 2.

A current direction can be specified in each phase.

Note1: The zero-current detection mode is not available in input process 2 task.

Note2: PWM shift 2 can only be selected in 1-shunt current detection mode.

a. Input transformation

<Equation>

[VDC fixed-point transformation/store]

if (***[VExMODE]***<VDCSEL> = 0) ***[VExVDC]*** = [DC voltage] / 2

else ***[VExVDCL]*** = [DC voltage] / 2

if (***[VExMCTLF]***<SFT2ST> ≠ ***[VExMCTLF]***<SFT2STM>) END (Processing completed)

; When the carrier reverse state is different from the previous value, processing is completed.

[Current 1 read]

if (***[VExMCTLF]***<SFT2ST> = 0)

- if ([Current 1 phase information] = 1) ***[VExIAADC]*** = [Current 1]
- else if ([Current 1 phase information] = 2) ***[VExIBADC]*** = [Current 1]
- else if ([Current 1 phase information] = 3) ***[VExICADC]*** = [Current 1]

else

- if ([Current 2 phase information] = 1) ***[VExIAADC]*** = [Current 1]
- else if ([Current 2 phase information] = 2) ***[VExIBADC]*** = [Current 1]
- else if ([Current 2 phase information] = 3) ***[VExICADC]*** = [Current 1]

[Current 2 read]

if (***[VExMCTLF]***<SFT2ST> = 0)

- if ([Current 2 phase information] = 1) ***[VExIAADC]*** = [Current 2]
- else if ([Current 2 phase information] = 2) ***[VExIBADC]*** = [Current 2]
- else if ([Current 2 phase information] = 3) ***[VExICADC]*** = [Current 2]

else

- if ([Current 1 phase information] = 1) ***[VExIAADC]*** = [Current 2]
- else if ([Current 1 phase information] = 2) ***[VExIBADC]*** = [Current 2]
- else if ([Current 1 phase information] = 3) ***[VExICADC]*** = [Current 2]

[Current 3 read]

if (***[VExFMODE]***<IDMODE[1:0]> = 00 ; Only when 3-phase detection is performed.

- if ([Current 3 phase information] = 1) ***[VExIAADC]*** = [Current 3]
- else if ([Current 3 phase information] = 2) ***[VExIBADC]*** = [Current 3]
- else if ([Current 3 phase information] = 3) ***[VExICADC]*** = [Current 3]

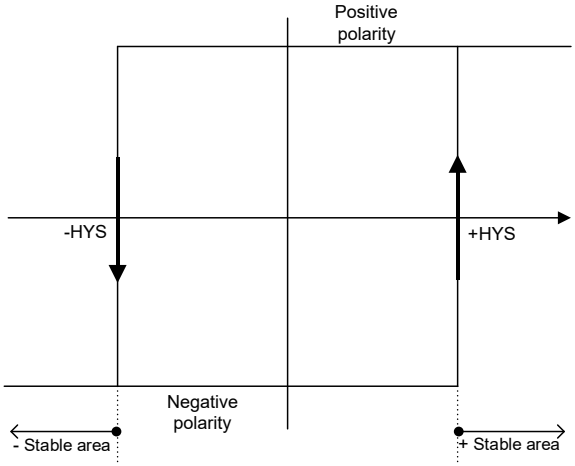
[Current fixed-point transformation]
ia = ***[VExIAO] - [VExIAADC]***
ib = ***[VExIBO] - [VExIBADC]***
ic = ***[VExICO] - [VExICADC]***
if (***[VExMCTLF]***<SFT2ST> = 0)
  if (***[VExFMODE]***<IAPLMD> = 1) ia = -ia ; Sets direction of Ia current detection
  if (***[VExFMODE]***<IBPLMD> = 1) ib = -ib ; Sets direction of Ib current detection
  if (***[VExFMODE]***<ICPLMD> = 1) ic = -ic ; Sets direction of Ic current detection
else
  if (***[VExFMODE]***<IAPLMD> = 0) ia = -ia ; Sets direction of Ia current detection
  if (***[VExFMODE]***<IBPLMD> = 0) ib = -ib ; Sets direction of Ib current detection
  if (***[VExFMODE]***<ICPLMD> = 0) ic = -ic ; Sets direction of Ic current detection
n = 6 - [Current 2 phase information] - [Current 1 phase information]
if (***[VExFMODE]***<IDMODE[1:0]> ≠ 00) ; Current 3 is calculated except when 3-phase detection is performed.
  if (n = 1) ia = -ib -ic
  else if (n = 2) ib = -ic -ia
  else if (n = 3) ic = -ia -ib

[Current storage]
[VExIA] = ia
[VExIB] = ib
[VExIC] = ic

	Register name/ signal name	Function	Detail
Input	ADxREG0	Input of the ADC conversion result	16-bit data (The conversion result is stored in the upper 12 bits.) ADxREG0: Current 1 input ADxREG1: Current 2 input ADxREG2: Current 3 input ADxREG3: DC voltage input
	ADxREG1		
	ADxREG2		
	ADxREG3		
	UVWIS0	Phase select information	UVWIS0 to 3 00: No select 01: Phase U 10: Phase V 11: Phase W
	UVWIS1		
	UVWIS2		
	UVWIS3		
	[VExMODE]	[4] Selects the supply voltage store	<VDCSEL> 0: Stored in ***[VExVDC]*** 1: Stored in ***[VExVDCL]***
	[VExFMODE]	[3:2] Selects current detection mode	<IDMODE[1:0]> 00: 3-shunt 01: 2 sensors 10, 11: 1-shunt
		[7:5] Select the current detection	<ICPLMD>, <IBPLMD>, <IAPLMD> 0: Shunt mode (In = VExInO - VExInADC) 1: Sensor mode (In = VExInADC - VExInO) Note: n = A, B, or C
	[VExMCTLF]	[14] PWM carrier reverse flag of PWM shift 2	<SFT2ST> 0: Without PWM carrier reverse of reference phase 1: With PWM carrier reverse of reference phase
		[15] previous value of <SFT2ST>	<SFT2STM>
	[VExIAO]	a-phase zero-current	16-bit data (The result is stored in the upper 12 bits.)
	[VExIBO]	b-phase zero-current	
	[VExICO]	c-phase zero-current	
Output	***[VExVDC]***	DC supply voltage	16-bit fixed-point data (0.0 to 1.0, 15 fractional bits)
	[VExVDCL]	DC2 supply voltage	
	[VExIAADC]	ADC conversion result for a-phase current	16-bit data (The result is stored in the upper 12 bits.)
	[VExIBADC]	ADC conversion result for b-phase current	
	[VExICADC]	ADC conversion result for c-phase current	
	[VExIA]	a-phase current	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExIB]	b-phase current	
	[VExIC]	c-phase current	

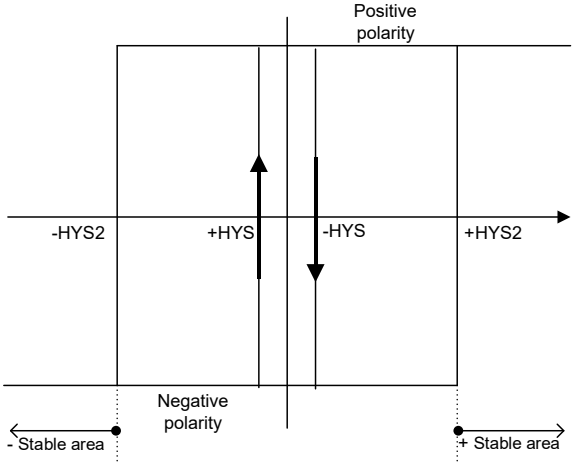
b. Current polarity determination

【**Hysteresis**: 0 ≥ HYS】



[I > HYS, I < -HYS]

【**Reverse hysteresis**: HYS < 0】



[|HYS2| ≤ I, I ≤ -|HYS2|, I > HYS, I < -HYS]

	Register name	Function	Detail
Input	***[VExHYS]***	Hysteresis width for current polarity determination	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExHYS2]	Hysteresis width 2 for current polarity determination	Polarity determination level on reverse hysteresis (***[VExHYS]*** < 0). 16-bit fixed-point data (0 to 1.0, 15 fractional bits) This register is invalid when ***[VExHYS]*** ≥ 0.
	[VExDTCS]	Dead time compensation control/status	<ICSTS[2:0]>, <IBSTS[2:0]>, <IASTS[2:0]> 000, 010, 100, 110: Polarity determination is undefined. Dead time compensation is not performed. 001, 101: Positive current is determined. Adds the ***[VExDTC]*** value when dead time compensation is performed. 011, 111: Negative current is determined. Subtracts the ***[VExDTC]*** value when dead time compensation is performed.
	[VExMODE]	[15] Disables/enables current polarity determination.	<IPDEN> 0: Disabled. 1: Enabled.
	[VExFMODE]	[9] Enables/disables the use of previous value flags of ***[VExMCTLF]***.	<MREGDIS> 0: Enabled 1: Disabled
	[VExIA]	a-phase current	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExIB]	b-phase current	
	[VExIC]	c-phase current	
Output	***[VExDTCS]***	Dead time compensation control/status	<ICSTS[2:0]>, <IBSTS[2:0]>, <IASTS[2:0]> 000, 010, 100, 110: Polarity determination is undefined. Dead time compensation is not performed. 001, 101: Positive current is determined. Adds the ***[VExDTC]*** value when dead time compensation is performed. 011, 111: Negative current is determined. Subtracts the ***[VExDTC]*** value when dead time compensation is performed.

3.3.7. Input Current Transformation (Phase Transformation/Coordinate Axis Transformation)

Input current transformation consists of two tasks: phase transformation and coordinate axis transformation.

3.3.7.1. Input Phase Transformation (Task 3)

Input phase transformation task calculates Iα and Iβ based on Ia, Ib, and Ic.

<Equation>

kc = 1 ; Relative transformation
if (***[VExFMODE]***<CCVMD> = 1) kc = √ (3 / 2) ; Absolute transformation
if (***[VExFMODE]***<PHCVDIS> = 0) ; Phase transformation is enabled
[VExIALPHA] = kc × ***[VExIA]*** - ***([VExIB] + [VExIC]) / 2*** ; Calculates the value of I_α
[VExIBETA] = kc × √ (1 / 3) × ***[VExIB]*** - kc × √ (1 / 3) × ***[VExIC]*** ; Calculates the value of I_β
else if (***[VExFMODE]***<PHCVDIS> = 1) ; Phase transformation is disabled
[VExIALPHA] = ***[VExIA]***
[VExIBETA] = ***[VExIB]***

	Register name	Function	Detail
Input	***[VExIA]***	a-phase current	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExIB]	b-phase current	
	[VExIC]	c-phase current	
	[VExFMODE]	[12] Disables/enables phase transformation.	<PHCVDIS> 0: 2-3 phase transformation is enabled. (3-phase AC output) 1: 2-3 phase transformation is disabled. (2-phase AC output)
		[13] Selects a phase transformation mode.	<CCVMD> 0: Relative transformation 1: Absolute transformation
Output	***[VExIALPHA]***	α-axis current	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExIBETA]	β-axis current	

3.3.7.2. Input Coordinate Axis Transformation (Task 4)

In input coordinate axis transformation task, the values of ***[VExID]*** and ***[VExIQ]*** are calculated from ***[VExIALFA]***, ***[VExIBETA]***, ***[VExSINM]***, and ***[VExCOSM]***.

a. Coordinate axis transformation

<Equation>

fmodf = (***[VExFMODEM]*** & 0xC002) – (***[VExFMODE]*** & 0xC002)

if (fmodf = 0) ; Shift mode change confirmation

[VExID] = ***[VExCOSM]*** × ***[VExIALPHA]*** + ***[VExSINM]*** × ***[VExIBETA]*** ; Calculates the value of Id.

[VExIQ] = -***[VExSINM]*** × ***[VExIALPHA]*** + ***[VExCOSM]*** × ***[VExIBETA]*** ; Calculates the value of Iq.

[VExFMODEM] = ***[VExFMODE]***

	Register name	Function	Detail
Input	***[VExIALPHA]***	α-axis current	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExIBETA]	β-axis current	
	[VExSINM]	Previous sine value	16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits)
	[VExCOSM]	Previous cosine value	
	[VExFMODE]	[1] Disables/enables PWM shift	<SPWMEN> 0: Disabled 1: Enabled
		[9] Enables/disables the use of previous value flags of ***[VExMCTLF]***.	<MREGDIS> 0: Enabled 1: Disabled
		[15:14] Selects a PWM shift mode	<SPWMMD[1:0]> 00: shift1 01: shift 2 (U-phase center) 10: shift 2 (V-phase center) 11: shift 2 (W-phase center)
Output	***[VExID]***	d-axis current	32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits)
	[VExIQ]	q-axis current	
	[VExFMODEM]	Flow control previous value	16-bit data

Note: Refer to "4.3.3. ***[VExFMODE]*** (Flow Control Register)" for bit details of ***[VExFMODEM]***.

b. ATAN calculation

When <ATANMD> = 10, ATAN task calculates the declination angle of d-axis and q-axis current.

When <ATANMD> = 11, ATAN task calculates the induced voltage of d-axis and q-axis. Then it calculates the declination angle using motor voltage equation based on these results.

<Equation>

```
if (ExMODE<ATANMD[1]> = 1)                                  ; Enables calculation of the declination angle
  ld = ExCLD × (ExCLG setting)                              ; d-axis inductance
  lq = ExCLQ × (ExCLG setting)                              ; q-axis inductance
  r = ExCR × (ExCRG setting)                                ; Resistance
  vdiv = ExVD - (ExID × r - ExOMEGA × ExIQ × lq)            ; Induced voltage for d-axis
  vqiv = ExVQ - (ExIQ × r + ExOMEGA × ExID × ld);           ; Induced voltage for q-axis
    ExEDDELTA = vdiv                                        ; Stores d-axis induced voltage
    ExEQDELTA = vqiv                                        ; Stores q-axis induced voltage
  if(ExMODE<ATANMD[0]> = 1) z = ATAN(|vqiv|, |vdiv|)        ; Calculates the declination angle of the induced voltage
  else z = ATAN(|ExIQ|, |ExID|)                             ; Calculates the declination angle of the current.
  if ((x < 0) && (y ≥ 0)) z = 0x00008000 - z                ; The second quadrant (90 to 180 degrees)
  if ((x < 0) && (y < 0)) z = 0xFFFF8000 + z                ; The third quadrant (-90 to -180 degrees)
  if ((x ≥ 0) && (y < 0)) z = - z                           ; The forth quadrant (0 to -90 degrees)
  if ((x = = 0) && (y = = 0)) z = 0x00000000                ; Output on the original point (0 degree)
  ExDELTA = z
```

*The previous value is automatically used for ***[VExIQ]*** and ***[VExID]*** if not update "a. coordinate axis transformation".

	Register name	Function	Detail
Input	***[VExCLD]***	Motor d-axis inductance value	16-bit fixed-point data (-16.0 to 16.0 and 11 fractional bits)
	[VExCLQ]	Motor q-axis inductance value	
	[VExCR]	Motor resistance value	
	[VExCLG]	Motor inductance range setting	000: 1 / 1 001: 1 / 2^4 010: 1 / 2^8 011: 1 / 2^{12} 100: 1 / 2^{16} 101 to 111: Reserved
	[VExCRG]	Motor resistance range setting	
	[VExOMEGA]	Rotational speed setting	16-bit fixed-point data (-1.0 to 1.0 and 15 fractional bits)
	[VExVD]	d-axis voltage	32-bit fixed-point data (-1.0 to 1.0 and 31 fractional bits)
	[VExVQ]	q-axis voltage	
	[VExMODE]	[6:5] Specifies ATAN calculation control.	<ATANMD[1:0]> 00, 01: Calculation is disabled. 10: Calculates the declination angle on d-q coordinate of current vector. 11: Calculates the declination angle on d-q coordinate of induced voltage vector.
Output	***[VExDELTA]***	Sets the current declination angle on the d-q coordinate.	16-bit data (-180 to 180 degrees) "0x8000" to "0x7FFF"
	[VExEDDELTA]	d-axis induced voltage	32-bit fixed-point data (-1.0 to 1.0 and 31 fractional bits)
	[VExEQDELTA]	q-axis induced voltage	

3.3.8. Other Tasks

3.3.8.1. ATAN2 (Arctangent Function 2) (Task 12)

ATAN2 task calculates the angle from the starting point based on X-axis that draws a straight line from the starting point to the point of (X, Y) on the X-Y coordinate.

<Equation>

x = ***[VExATANX]***
y = ***[VExATANY]***
z = ATAN (|y|, |x|) ; Arctangent calculation (0 to 90 degrees)
if ((x < 0) && (y ≥ 0)) z = 0x8000 - z ; The second quadrant (90 to 180 degrees)
if ((x < 0) && (y < 0)) z = 0x8000 + z ; The third quadrant (-90 to -180 degrees)
if ((x ≥ 0) && (y < 0)) z = - z ; The forth quadrant (0 to -90 degrees)
if ((x = = 0) && (y = = 0)) z = 0x0000 ; Output on the original point (0 degree)
[VEx ATANTHETA] = z

	Register name	Function	Detail
Input	***[VExATANX]***	ATAN2 function X input	32-bit fixed-point data (-1.0 to 1.0 and 31 fractional bits)
	[VExATANY]	ATAN2 function Y input	
Output	***[VExATANTHETA]***	ATAN2 function phase difference	16-bit fixed-point data (-1.0 to 1.0 and 15 fractional bits) (-180 to 180 degrees)

3.3.8.2. SQRT (Square Root Function) (Task 13)

SQRT task outputs the value from 0.0 to 2.0 based on the calculation of the square root function using the inputs from 0.0 to 4.0.

<Equation>

x = ***[VExSQRTIN]*** ; Input (0 to 4.0)
n = 0
if (x < 0x2000) n = 1
if (x < 0x0800) n = 2
if (x < 0x0200) n = 3
if (x < 0x0080) n = 4
if (x < 0x0020) n = 5
if (x < 0x0008) n = 6
if (x < 0x0002) n = 7
if (x < 0x0001) n = 8
if (x < 0x8000) n = -1
$x = x \times 2^{2n}$; Normalization (0.25 to 1.0)
if (x ≥ 0x7FFF) x = 0x7FFF
z = SQRT (x) ; Calculates the square root of x,
; Output is the range of 0.5 to 1.0
if (x = 0) z = 0
$z = z / 2^{n}$; Inverted transformation (0 to 2.0)
[VExSQRTOUT] = z

	Register name	Function	Detail
Input	***[VExSQRTIN]***	SQRT function input	32-bit fixed-point data (0.0 to 4.0, 15 fractional bits) "0x00000000" to "0x0001FFFF"
Output	***[VExSQRTOUT]***	SQRT function output	32-bit fixed-point data (0.0 to 2.0, 15 fractional bits) "0x00000000" to "0x00010000"

3.3.8.3. NOP Execution (Task 239)

It is used when to pause the processing after the output control and wait until the AD conversion is completed.

3.4. Debug Output

The VE task transition timing can be output via PMD. The VE task transition timing can be monitored by setting ***[PMDxDBGOUTCR]***<DBGMD[1:0]> = 11 and ***[PMDxDBGOUTCR]***<VEENDBGSEL> = 0. The debug output is inverted every time the VE task transition timing is input.

3.5. VE ROM Check Function

The VE ROM check function is used for reading VE ROM.
When the VE ROM is required reading by the bus masters, set ***[VExROMCONFG]***<VEROMBUS> to "0".
At this time, confirm that the VE stops before setting ***[VExROMCONFG]***<VEROMBUS> to "0".
And, a clock for VE is required when the CPU accesses to the VE.
[VExROMCONFG]<VEROMBUS> can not be set in case that the SAM is in App state when it is enabled.

4. Registers

The vector engine has the VE control registers and the dedicated registers.

- VE control registers
  The vector engine control registers and temporary registers.

- Dedicated registers
  Computation data registers and computation control registers.

Note: Use word access (32 bits) to the registers.

4.1. List of Registers

The control registers and their addresses are shown as follows:

Peripheral function		Channel/unit	Base address		
			TYPE1	TYPE2	TYPE3
Advanced Vector Engine 2	A-VE2	ch0	0x400F8000	0x400EB000	0x4008B000

Note: The base address type is different by products. Please refer to "Product Information" of the reference manual for the details.

4.1.1. VE Control Register

Register name		Address (Base+)
VE Operation Enable/Disable Register	***[VExEN]***	0x0000
CPU Start Trigger Selection Register	***[VExCPURUNTRG]***	0x0004
Operation Schedule Repeat Number Selection Register	***[VExREPTIME]***	0x0010
Start Trigger Mode Setting Register	***[VExTRGMODE]***	0x0014
Error Interrupt Enable/Disable Setting Register	***[VExERRINTEN]***	0x0018
VE Forcible Termination Register	***[VExCOMPEND]***	0x001C
Error Detection Register	***[VExERRDET]***	0x0020
Schedule Operation Status/Ongoing Task Number Register	***[VExSCHTASKRUN]***	0x0024
Temporary Register 0	***[VExTMPREG0]***	0x002C
Temporary Register 1	***[VExTMPREG1]***	0x0030
Temporary Register 2	***[VExTMPREG2]***	0x0034
Temporary Register 3	***[VExTMPREG3]***	0x0038
Temporary Register 4	***[VExTMPREG4]***	0x003C
Temporary Register 5	***[VExTMPREG5]***	0x0040
Program Task Pointer Start Register	***[VExTSKINITP]***	0x0280
Program Task Register n (n = 0 to 31)	***[VExTSKPGMn]***	0x0284 to 0x0300
SRAM Task Setting Register 1	***[VExRAMTSKCONFG1]***	0x0308
VE ROM Setting Register	***[VExROMCONFG]***	0x030C

4.1.2. Dedicated Registers

Register name		Address (Base+)
Status Register	***[VExMCTLF]***	0x0044
Task Control Mode Register	***[VExMODE]***	0x0048
Flow Control Register	***[VExFMODE]***	0x004C
PWM Period Rate Setting Register	***[VExTPWM]***	0x0050
Rotational Speed Setting Register	***[VExOMEGA]***	0x0054
Motor Phase Setting Register	***[VExTHETA]***	0x0058
d-axis Current Reference Setting Register	***[VExIDREF]***	0x005C
q-axis Current Reference Setting Register	***[VExIQREF]***	0x0060
d-axis Voltage Register	***[VExVD]***	0x0064
q-axis Voltage Register	***[VExVQ]***	0x0068
d-axis Current Integral Coefficient Setting Register for PI Control	***[VExCIDKI]***	0x006C
d-axis Current Proportional Coefficient Setting Register for PI Control	***[VExCIDKP]***	0x0070
q-axis Current Integral Coefficient Setting Register for PI Control	***[VExCIQKI]***	0x0074
q-axis Current Proportional Coefficient Setting Register for PI Control	***[VExCIQKP]***	0x0078
d-axis Voltage Integral Coefficient Store Register for PI Control (Upper 32 bits)	***[VExVDIH]***	0x007C
d-axis Voltage Integral Coefficient Store Register for PI Control (Lower 16 bits)	***[VExVDILH]***	0x0080
q-axis Voltage Integral Coefficient Store Register for PI Control (Upper 32 bits)	***[VExVQIH]***	0x0084
q-axis Voltage Integral Coefficient Store Register for PI Control (Lower 16 bits)	***[VExVQILH]***	0x0088
PWM Switching Speed Setting Register	***[VExFPWMCHG]***	0x008C
PWM Shift 2 Offset Register	***[VExPWMOFS]***	0x0090
Synchronous Trigger Correction Setting Register	***[VExTRGCRC]***	0x0098
DC2 Supply Voltage Register	***[VExVDCL]***	0x009C
Cosine Value Register of θ	***[VExCOS]***	0x00A0
Sine Value Register of θ	***[VExSIN]***	0x00A4
Previous Cosine Value Register	***[VExCOSM]***	0x00A8
Previous Sine Value Register	***[VExSINM]***	0x00AC
Sector Information Register	***[VExSECTOR]***	0x00B0
Previous Sector Information Register	***[VExSECTORM]***	0x00B4
a-phase Zero-current Register	***[VExIAO]***	0x00B8
b-phase Zero-current Register	***[VExIBO]***	0x00BC
c-phase Zero-current Register	***[VExICO]***	0x00C0
ADC Conversion Result Register for a-phase Current	***[VExIAADC]***	0x00C4
ADC Conversion Result Register for b-phase Current	***[VExIBADC]***	0x00C8
ADC Conversion Result Register for c-phase Current	***[VExICADC]***	0x00CC
DC Supply Voltage Register	***[VExVDC]***	0x00D0
d-axis Current Register	***[VExID]***	0x00D4
q-axis Current Register	***[VExIQ]***	0x00D8
d-axis Induced Voltage (Declination Output) Register	***[VExEDDELTA]***	0x00DC
q-axis Induced Voltage (Declination Output) Register	***[VExEQDELTA]***	0x00E0
Flow Control Previous Value Register	***[VExFMODEM]***	0x00F8
ADC Conversion Time Setting Register	***[VExTADC]***	0x0178
U-phase PWM Duty Register	***[VExCMPU]***	0x017C

Register name		Address (Base+)
V-phase PWM Duty Register	***[VExCMPV]***	0x0180
W-phase PWM Duty Register	***[VExCMPW]***	0x0184
PMD Output Control Register	***[VExOUTCR]***	0x0188
PMD Trigger Timing Setting Register 0	***[VExTRGCMP0]***	0x018C
PMD Trigger Timing Setting Register 1	***[VExTRGCMP1]***	0x0190
Synchronous Trigger Selection Setting Register	***[VExTRGSEL]***	0x0194
Zero Vector Duty Register	***[VExDUTYZERO]***	0x0198
ATAN2 Function X Input Register	***[VExATANX]***	0x019C
ATAN2 Function Y Input Register	***[VExATANY]***	0x01A0
SQRT Function Input Register	***[VExSQRTIN]***	0x01A4
SQRT Function Output Register	***[VExSQRTOUT]***	0x01A8
Output Limitation Register for PI Control	***[VExPIOLIM]***	0x01BC
d-axis Coefficient Range Setting Register for PI Control	***[VExCIDKG]***	0x01C0
q-axis Coefficient Range Setting Register for PI Control	***[VExCIQKG]***	0x01C4
Voltage Scalar Limitation Register	***[VExVSLIM]***	0x01C8
Voltage Scalar Register	***[VExVDQ]***	0x01CC
Declination Angle Register	***[VExDELTA]***	0x01D0
Motor Interlinkage Magnetic Flux Register	***[VExCPHI]***	0x01D4
Motor d-axis Inductance Value Register	***[VExCLD]***	0x01D8
Motor q-axis Inductance Value Register	***[VExCLQ]***	0x01DC
Motor Resistance Value Register	***[VExCR]***	0x01E0
Motor Magnetic Flux Range Setting Register	***[VExCPHIG]***	0x01E4
Motor Inductance Range Setting Register	***[VExCLG]***	0x01E8
Motor Resistance Range Setting Register	***[VExCRG]***	0x01EC
d-axis Voltage Register for Non-interference Control	***[VExVDE]***	0x01F0
q-axis Voltage Register for Non-interference Control	***[VExVQE]***	0x01F4
Dead Time Compensation Register	***[VExDTC]***	0x01F8
Hysteresis Width Register for Current Polarity Determination	***[VExHYS]***	0x01FC
Dead Time Compensation Control/Status Register	***[VExDTCS]***	0x0200
PWM Upper Limit Setting Register	***[VExPWMMAX]***	0x0204
PWM Lower Limit Setting Register	***[VExPWMMIN]***	0x0208
Clipped Phase Value Setting Register	***[VExTHTCLP]***	0x020C
Hysteresis Width Register 2 for Current Polarity Determination	***[VExHYS2]***	0x0210
α- axis Voltage Register	***[VExVALPHA]***	0x0214
β- axis Voltage Register	***[VExVBETA]***	0x0218
a-phase Voltage Duty Register	***[VExVDUTYA]***	0x021C
b-phase Voltage Duty Register	***[VExVDUTYB]***	0x0220
c-phase Voltage Duty Register	***[VExVDUTYC]***	0x0224
α- axis Current Register	***[VExIALPHA]***	0x0228
β- axis Current Register	***[VExIBETA]***	0x022C
a-phase Current Register	***[VExIA]***	0x0230
b-phase Current Register	***[VExIB]***	0x0234
c-phase Current Register	***[VExIC]***	0x0238
Voltage Declination Angle Register	***[VExVDELTA]***	0x023C
d-axis Voltage Correction Register	***[VExVDCRC]***	0x0240
q-axis Voltage Correction Register	***[VExVQCRC]***	0x0244
ATAN2 Function Phase Difference Register	***[VExATANTHETA]***	0x0248

4.2. Details of VE Control Registers

4.2.1. *[VExEN]* (VE Operation Enable/Disable Register)

Bit	Bit symbol	After reset	Type	Function
31:2	-	0	R	Read as "0".
1	-	0	R/W	Write as "0".
0	VEEN	0	R/W	Selects the VE operation 0: Disabled 1: Enabled

Note: When the VE is disabled (<VEEN> = 0), access to other registers of the VE is not allowed

Note: To change the PMD registers from VE, be sure to set the operation schedule repetition count specification register ***[VExREPTIME]***<VREP> to other than "0x0" after enabling the VE operation (<VEEN> = 1).

4.2.2. *[VExCPURUNTRG]* (CPU Start Trigger Selection Register)

Bit	Bit symbol	After reset	Type	Function
31:1	-	0	R	Read as "0".
0	VCPURT	0	W	Starts the vector engine by software. 0: - 1: Starts operation Operation starts from the task configured to ***[VExTSKINITP]***<PTR>. Specify ***[VExTSKINITP]*** and ***[VExREPTIME]*** before starting operation.

Note: When "1" is written to this bit, it is cleared on the next cycle. This bit always read as "0".

Note: If new schedules and the tasks start operation while another schedule is being executed, the vector engine should be terminated with the ***[VExCOMPEND]*** register before it is started with the ***[VExCPURUNTRG]*** register again.

4.2.3. *[VExTSKINITP]* (Program Task Pointer Start Register)

Bit	Bit symbol	After reset	Type	Function
31:5	-	0	R	Read as "0".
4:0	PTR[4:0]	00000	R/W	Start task pointer Specifies pointer of first task that the scheduler.

4.2.4. *[VExTSKPGMn]* (Program Task Register n (n = 0 to 31)

Example of ***[VExTSKPGM0]. [VExTSKPGM1] to [VExTSKPGM31]*** have the same configuration.

Bit	Bit symbol	After reset	Type	Function
31:11	-	0	R	Read as "0".
10	END	0	R/W	End task specification 0: After the task specified by CODE is completed, the next task will be scheduled. 1: After the task specified by CODE is completed, the scheduler will end.
9	ADC	0	R/W	AD conversion completion trigger wait task specification 0: After the task specified by CODE is completed, the next task will be scheduled. 1: After the task specified by CODE is completed, the ADC interrupt is waited.
8	-	0	R/W	Write as "0".
7:0	CODE[7:0]	0x00	R/W	Specify the task type 0x00: Output control 1 (Task 0) 0x01: Trigger generation (Task 1) 0x02: Input process 1 (Task 2) 0x03: Input phase transformation (Task 3) 0x04: Input coordinate axis transformation (Task 4) 0x05: Current control (Task 5) 0x06: SIN/COS calculation (Task 6) 0x07: Output coordinate axis transformation (Task 7) 0x08: Output phase transformation 1 [SVM] (Task 8) 0x09: Output control 2 (Task 9) 0x0A: Input process 2 (Task 10) 0x0B: Output phase transformation 2 [I-Clark] (Task 11) 0x0C: ATAN2 (Task 12) 0x0D: SQRT (Task 13) 0x0E to 0xEE: Reserved 0xEF: NOP (No operation) (Task 239) 0xFF: Reserved

4.2.5. *[VExREPTIME]* (Operation Schedule Repeat Number Selection Register)

Bit	Bit symbol	After reset	Type	Function
31:4	-	0	R	Read as "0".
3:0	VREP[3:0]	0x0	R/W	Selects the number of repeat of the operation schedule. 0x0: No schedule operation is performed. 0x1 to 0xF: Repeats the specified number of times of the schedule operation

Note: When this bit is set to "0x0", do not start schedule operation.

4.2.6. *[VExTRGMODE]* (Start Trigger Mode Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:2	-	0	R	Read as "0".
1:0	VTRG[1:0]	00	R/W	Selects the input process start condition using an ADC conversion end interrupt. 00: Reserved 01: Input process is started by a INTADxPDA (ADC conversion end interrupt A). (Note) 10: Input process is started by a INTADxPDB (ADC conversion end interrupt B). (Note) 11: Reserved

Note: Connection varies depending on the product. Refer to "Product Information".

4.2.7. *[VExERRINTEN]* (Error Interrupt Enable/Disable Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:1	-	0	R	Read as "0".
0	VERREN	0	R/W	Controls an interrupt at error detection. 0: Disabled 1: Enabled If a PWM interrupt is detected when an operation schedule is being executed (an activation trigger wait is not included), "1" is set as an error flag.

4.2.8. *[VExCOMPEND]* (VE Forcible Termination Register)

Bit	Bit symbol	After reset	Type	Function
31:1	-	0	R	Read as "0".
0	VCEND	0	W	Ongoing schedule is forcibly terminated. 0: - 1: Stop If "1" is written to this bit, this bit is cleared on next cycle. Always read as "0".

4.2.9. *[VExERRDET]* (Error Detection Register)

Bit	Bit symbol	After reset	Type	Function
31:1	-	0	R	Read as "0".
0	VERRD	0	R	This bit indicates the error flag. 0: An error is not detected. 1: An error is detected. When an operation schedule is being executed (excluding a start trigger wait), if a PWM interrupt is detected, "1" is set to this bit. This bit is cleared upon read.

4.2.10. *[VExSCHTASKRUN]* (Schedule Operation Status/Ongoing Task Number Register)

Bit	Bit symbol	After reset	Type	Function
31:12	-	0	R	Read as "0".
11:4	VRTASK[7:0]	0x00	R	Indicates the ongoing task number. 0x00: Output control 1 (Task 0) 0x01: Trigger generation (Task 1) 0x02: Input process 1 (Task 2) 0x03: Input phase transformation (Task 3) 0x04: Input coordinate axis transformation (Task 4) 0x05: Current control (Task 5) 0x06: SIN/COS calculation (Task 6) 0x07: Output coordinate axis transformation (Task 7) 0x08: Output phase transformation 1[SVM] (Task 8) 0x09: Output control 2 (Task 9) 0x0A: Input process 2 (Task 10) 0x0B: Output phase transformation 2[I-Clark] (Task 11) 0x0C: ATAN2 (Task 12) 0x0D: SQRT (Task 13) 0x0E to 0xEE: Reserved 0xEF: NOP (No operation) (Task 239) 0xF0 to 0xFF: Reserved
3:1	-	0	R	Read as "0".
0	VRSCH	0	R	Indicates schedule execution status. 0: Stop 1: Being executed

4.2.11. *[VExTMPREG0]* (Temporary Registers 0)

Example of ***[VExTMPREG0]***. ***[VExTMPREG1]*** to ***[VExTMPREG5]*** have the same configuration.

Bit	Bit symbol	After reset	Type	Function
31:0	TMPREG0 [31:0]	0x00000000	R/W	Temporary register 0

4.2.12. *[VExRAMTSKCONFG1]* (SRAM Task Setting Register 1)

Bit	Bit symbol	After reset	Type	Function
31	VERAMBUS	0	R/W	Write as "0".
30	-	0	R	Read as "0".
29:24	-	0x00	R/W	Write as "0".
23:22	-	00	R	Read as "0".
21:16	-	0x00	R/W	Write as "0".
15:14	-	00	R	Read as "0".
13:8	-	0x00	R/W	Write as "0".
7:6	-	00	R	Read as "0".
5:0	-	0x00	R/W	Write as "0".

4.2.13. *[VExROMCONFG]* (VE ROM Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:1	-	0	R	Read as "0".
0	VEROMBUS	1	R/W	Bus selection to connect to VE ROM area 0: AHB bus connection (CPU access) 1: VE bus connection ***[VExROMCONFG]***<VEROMBUS> can not be set in case that the SAM is in App state when it is enabled.

4.3. Details of Dedicated Registers

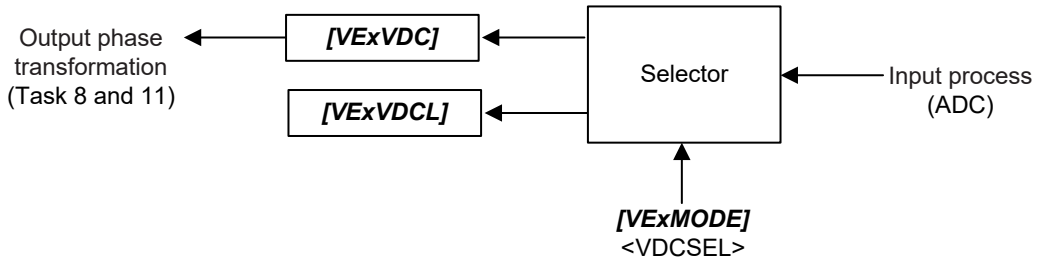
4.3.1. *[VExMCTLF]* (Status Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15	SFT2STM	0	R/W	Previous value of <SFT2ST> Updated when output control 1, 2 (Task 0, 9) is executed. Used for input processing 2 (Task 10).
14	SFT2ST	0	R/W	A PWM carrier reverse flag of PWM shift 2 0: Without PWM carrier reverse of reference phase 1: With PWM carrier reverse of reference phase Set "1" to this bit when Shift 2 reference phase duty < 0x4000 Updated when output control 1, 2 (Task 0, 9) is executed. Used for input processing 2 (Task 10).
13:12	-	00	R/W	Write as "00".
11	PWMOVF	0	R/W	Indicates the excess flag for PWM output limitation. 0: All 3-phase PWM outputs are ***[VExPWMMIN]*** or more, and ***[VExPWMMAX]*** or less. 1: Any 3-phase PWM outputs are less than ***[VExPWMMIN]***, or over ***[VExPWMMAX]***. This bit is updated when output controls (Task 0, 9) are executed.
10	VSOVF	0	R/W	Indicates the excess flag for voltage scalar limitation. 0: Voltage scalar ≤ ***[VExVSLIM]*** 1: Voltage scalar > ***[VExVSLIM]*** This bit is updated when current control (Task 5) is executed.
9	PIQOVF	0	R/W	Indicates the excess flag for q-axis output limitation on PI control. 0: |q-axis output on PI control | ≤ ***[VExPIOLIM]*** 1: |q-axis output on PI control | > ***[VExPIOLIM]*** This bit is updated when current control (Task 5) is executed.
8	PIDOVF	0	R/W	Indicates the excess flag for d-axis output limitation on PI control. 0: |d-axis output on PI control | ≤ ***[VExPIOLIM]*** 1: |d-axis output on PI control | > ***[VExPIOLIM]*** This bit is updated when current control (Task 5) is executed.
7:3	-	0	R/W	Write as "0".
2	LVTF	0	R/W	Low-supply voltage flag 0: ***[VExVDC]*** ≥ 1 / 128 (0x0100) 1: ***[VExVDC]*** < 1 / 128 (0x0100) This bit is updated when output phase transformations (Task 8 , 11) are executed.
1	LAVFM	0	R/W	Previous value of the <LAVF> register This bit is updated when output controls (Task 0, 9) are executed. Used for input processing 1 (Task 2).
0	LAVF	0	R/W	Low-speed flag This bit is updated when output controls (Task 0, 9) are executed. 0: High-speed (***[VExOMEGA]*** ≥ ***[VExFPWMCHG]***) 1: Low-speed (***[VExOMEGA]*** < ***[VExFPWMCHG]***) Updated when output control 1, 2 (Task 0, 9) is executed. Used for trigger generation (Task 1) and input processing 1 (Task 2).

4.3.2. *[VExMODE]* (Task Control Mode Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15	IPDEN	0	R/W	Disables/enables current polarity determination for Task 2 and Task 10. 0: Disabled 1: Enabled Refer to "3.3.6 Input Process".
14	PMDDTCEN	0	R/W	Setting that corresponds to the dead time correction control of the PMD when the dead time compensation for Task 0 and Task 9 is performed. 0: Dead time correction of the PMD is disabled. 1: Dead time correction of the PMD is enabled. Refer to "3.3.4 Output Control".
13	PWMFLEN	0	R/W	Disables/enables a duty of 100% setting when the upper limit for Task 0 and Task 9 is set. 0: Duty of 100% setting is disabled. 1: Duty of 100% setting is enabled. Refer to "3.3.4 Output Control".
12	PWMBLEN	0	R/W	Disables/enables a duty of 0% setting when the lower limit for Task 0 and Task 9 is set. 0: Duty of 0% setting is disabled. 1: Duty of 0% setting is enabled. Refer to "3.3.4 Output Control".
11	NICEN	0	R/W	Disables/enables Non-interference control for Task 5. 0: Disabled 1: Enabled Refer to "3.3.1 Current Control (Task 5)".
10	T5ECEN	0	R/W	Disables/enables expansion control (Non-interference control and voltage scalar limitation) for Task 5. 0: Disabled 1: Enabled Refer to "3.3.1 Current Control (Task 5)".
9:8	AWUMD[1:0]	00	R/W	Specifies anti-windup (AWU) control for Task 5 when PI control output limitation is performed. 00: AWU control is disabled. 01: Substitutes the result from amount of limitation / 4 into the integral term. 10: Substitutes the result from amount of limitation / 2 into the integral term. 11: Substitutes the amount of limitation into the integral term. Refer to "3.3.1 Current Control (Task 5)".
7	CLPEN	0	R/W	Disables/enables phase clipping control for Task 6 when phase interpolation is performed. 0: Clipping is disabled. 1: Clipping is enabled. Refer to "3.3.2 SIN/COS Calculation (Task 6)".

Bit	Bit symbol	After reset	Type	Function
6:5	ATANMD [1:0]	00	R/W	Specifies ATAN calculation control for Task 4. 00, 01: Calculation is disabled. 10: Calculates the declination angle on d-q coordinate of current vector. 11: Calculates the declination angle on d-q coordinate of induced voltage vector. Refer to "3.3.7 Input Current Transformation (Phase Transformation/Coordinate Axis Transformation)".
4	VDCSEL	0	R/W	Selects the supply voltage store register for Task 2 and Task 10. 0: Stored in ***[VExVDC]***. 1: Stored in ***[VExVDCL]***. Refer to Figure 4.1.
3:2	OCRMD[1:0]	00	R/W	Specifies output control for Task 0, Task 1 and Task 9. 00: Output is disabled. 01: Output is enabled. 10: Short circuit brake (Outputs are off in the upper-phase; outputs are on in the lower-phase.) 11: Reserved Refer to "3.3.4 Output Control".
1	ZIEN	0	R/W	Specifies zero-current detection control for Task 1 and Task 2. 0: Normal current detection 1: Zero-current detection Refer to "3.3.6 Input Process".
0	PVIEN	0	R/W	Disables/enables phase interpolation control for Task 6. 0: Disabled 1: Enabled Refer to "3.3.2 SIN/COS Calculation (Task 6)".



Note: When a power supply voltage controlled by ***[VExVDC]*** register is corrected, select ***[VExVDCL]*** register as a storage location, and set a corrected value in ***[VExDVC]*** register.

Figure 4.1 *[VExVDC]*/*[VExVDCL]* Store Register

4.3.3. *[VExFMODE]* (Flow Control Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:14	SPWMMD[1:0]	00	R/W	Selects a PWM shift mode. 00: Shift 1 01: Shift 2 (U-phase center) 10: Shift 2 (V-phase center) 11: Shift 2 (W-phase center) Shift 2 cannot be selected when output control 1 (Task 0) is being executed (invalid). Used for output control 2 (Task 9), trigger generation (Task 1), and input coordinate axis transformation (Task 4).
13	CCVMD	0	R/W	Selects a phase transformation mode. 0: Relative transformation 1: Absolute transformation Used for output phase transformation 1, 2 (Tasks 8, 11) and input phase transformation (Task 3).
12	PHCVDIS	0	R/W	Disables/enables phase transformation. 0: 2-3 phase transformation is enabled. (3-phase AC output) 1: 2-3 phase transformation is disabled. (2-phase AC output) Phase transformation cannot be disabled when space vector modulation (Task 8) is running (invalid). Used for output phase transformation 2 (Task 11) and input phase transformation (Task 3).
11:10	VSLIMMD[1:0]	00	R/W	Controls voltage scalar limitation of current control (Task 5). 00: Scalar limitation is disabled. (The limitation of each axis is enabled.) 01: Limits the voltage on the d-axis. 10: Limits the voltage on the q-axis. 11: dq proportional limitation (Note) Note: Use within the range of |***[VExVD]***| / |***[VExVQ]***| ≥ 1/32768.
9	MREGDIS	0	R/W	Enables/disables the use of previous value flags of ***[VExMCTLF]***, ***[VExSIN]***, ***[VExCOS]***, and ***[VExSECTOR]*** registers on the input process 1, 2, and on input coordinate transformation. 0: Enabled 1: Disabled Used for SIN/COS calculation 1 (Task 6), output phase transformation 1, 2 (Task 8, 11), input process 1, 2 (Task 2, 10), and input coordinate axis transformation (Task 4).
8	CRCEN	0	R/W	Disables/enables trigger correction 0: Disabled 1: Enabled This bit is enabled when trigger generation (Task 1) is being executed; when PWM shift is disabled in 1-shunt current detection mode; or when Shift 1 is selected.
7	ICPLMD	0	R/W	Selects the current direction (***[VExIC]***) of input process2 (Task 10). 0: Shunt mode (***[VExIC]*** = ***[VExICO]*** – ***[VExICADC]***) 1: Sensor mode (***[VExIC]*** = ***[VExICADC]*** – ***[VExICO]***)
6	IBPLMD	0	R/W	Selects the current direction (***[VExIB]***) of input process2 (Task 10). 0: Shunt mode (***[VExIB]*** = ***[VExIBO]*** – ***[VExIBADC]***) 1: Sensor mode (***[VExIB]*** = ***[VExIBADC]*** – ***[VExIBO]***)

Bit	Bit symbol	After reset	Type	Function
5	IAPLMD	0	R/W	Selects the current direction (***[VExIA]***) of input process2 (Task 10). 0: Shunt mode (***[VExIA]*** = ***[VExIAO]*** – ***[VExIAADC]***) 1: Sensor mode (***[VExIA]*** = ***[VExIAADC]*** – ***[VExIAO]***)
4	IDQSEL	0	R/W	Selects Non-interference control input current 0: Feedback current (***[VExID]***, ***[VExIQ]***) 1: Command current (***[VExIDREF]***, ***[VExIQREF]***)
3:2	IDMODE[1:0]	00	R/W	Selects current detection mode. 00: 3-shunt (Note 1) 01: 2-sensor (Note 2) 10: 1-shunt (PMD TRG in the latter half of PWM (Note 3)) 11: 1-shunt (PMD TRG in the first half of PWM (Note 3)) Note 1: 3-phase current detection is used in input process 2 (Task 10). Note 2: 2-phase current detection is used in input process 2 (Task 10). Note 3: PWM shift 2 should be used when output control 2 (Task 9) and input process 2 (Task 10) is being executed. Used for output control 1, 2 (Task 0, 9), trigger generation (Task 1), and input process 1, 2 (Task 2, 10).
1	SPWMEN	0	R/W	Disables/enables PWM shift. 0: Disabled 1: Enabled Output control 1 (Task 0) and input process 1 (Task 2) are available only in Shift 1 mode. Output control 2 (Task 9) and input process 2 (Task 10) are available only in Shift 2 mode. Used for output control 1 and 2 (Task 0, 9), trigger generation (Task 1), input process 1 (Task 2), and input coordinate axis transformation (Task 4).
0	C2PEN	0	R/W	Selects the modulation mode. 0: 3-phase modulation 1: 2-phase modulation Used for output phase transformation 1 (Task 8), output control 1 (Task 0), trigger generation (Task 1), and input process 1 (Task 2).

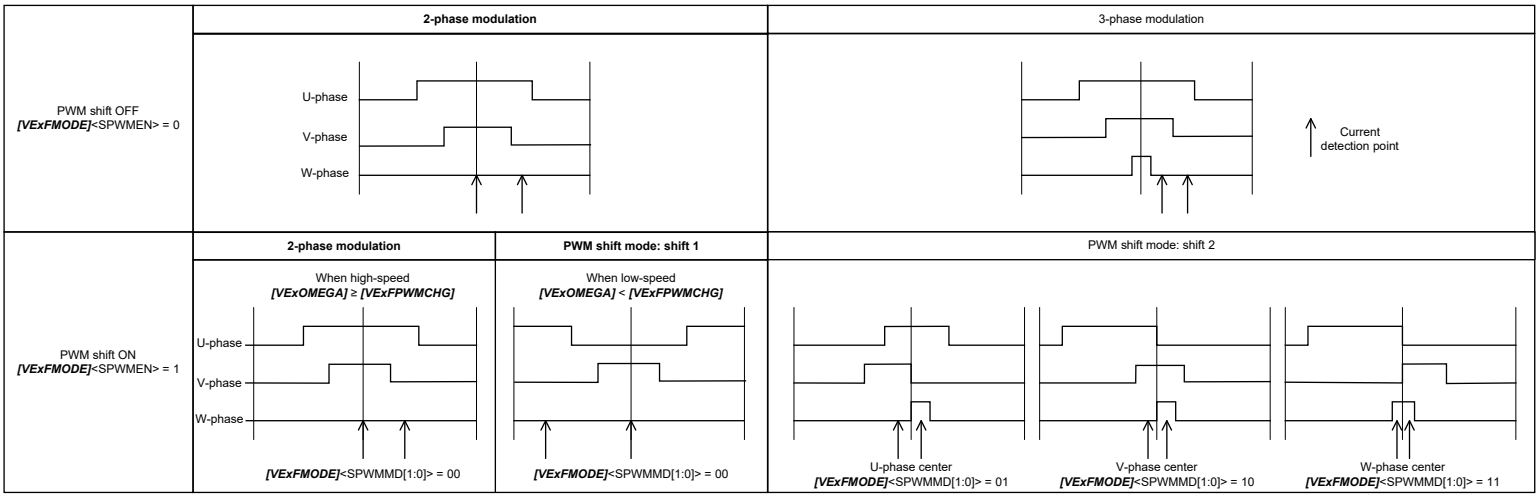


Figure 4.2 PWM Shift ON/OFF Waveform

Figure 4.3 shows the relationship between the normal PWM, PWM shift 1, and PWM shift 2 modulation methods and current detection rate. Current detection rate indicates the rate at which current can be detected per electrical angle. As for how to use it, it is assumed that a modulation method with a high current detection rate will be selected according to the modulation rate.

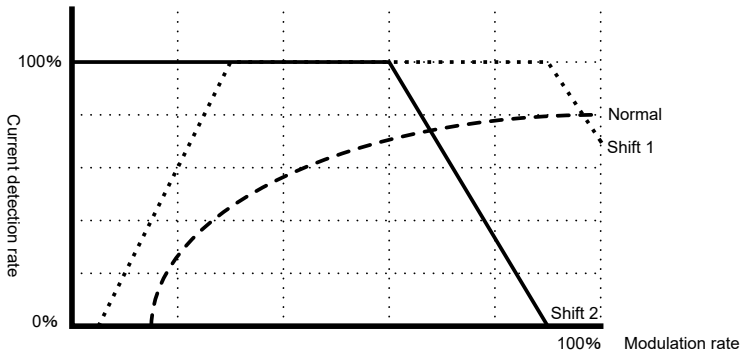


Figure 4.3 Each Modulation Method and Current Detection Rate

4.3.4. *[VExFMODEM]* (Flow Control Previous Value Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	FMODEM[15:0]	0x0000	R/W	Flow control (Previous value) 16-bit data Updated when input coordinate axis transformation (Task 4) is executed.

4.3.5. *[VExTPWM]* (PWM Period Rate Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	TPWM[15:0]	0x0000	R/W	Sets the PWM period rate and the unit of integration in phase interpolation (16-bit fixed-point data: 0.0 to 1.0, 15 fractional bits). "0x0000" to "0xFFFF" (PWM period[s] × Max_Hz × 2^{16}) This indicates the ratio between PWM period and the maximum rotational speed. (Max_Hz: Maximum rotational speed) This bit is used for SIN/COS calculation (Task 6) when phase interpolation is enabled.

4.3.6. *[VExOMEGA]* (Rotational Speed Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	OMEGA[15:0]	0x0000	R/W	Sets a rotational speed (16-bit fixed-point data: -1.0 to 1.0, 15 fractional bits). "0x8000" to "0x7FFF": Rotational speed [Hz] / Max_Hz × 2^{15} (Max_Hz: Maximum rotational speed) This bit is used for SIN/COS calculation (Task 6) when phase interpolation is enabled. This bit is used for output control 1 (Task 0) when PWM shift 1 is selected for 1-shunt current detection. This bit is used for current control (Task 5) and input coordinate axis transformation (Task 4) in motor voltage equation.

4.3.7. *[VExTHETA]* (Motor Phase Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	THETA[15:0]	0x0000	R/W	Sets the phase (16-bit fixed-point data: 0.0 to 1.0, 15 fractional bits). "0x0000" to "0xFFFF": Phase [deg] / 360 × 2^{16} This bit is used for SIN/COS calculation (Task 6) when phase interpolation is enabled. This bit is updated when SIN/COS calculation (Task 6) is executed while phase interpolation is enabled.

4.3.8. d-axis/q-axis Current Reference Setting Registers

4.3.8.1. *[VExIDREF]* (d-axis Current Reference Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	IDREF[15:0]	0x0000	R/W	Sets the reference value of d-axis current (16-bit fixed-point data: -1.0 to 1.0, 15 fractional bits). "0x8000" to "0x7FFF": d-axis current reference [A] / Max_I × 2^{15} Max_I: The amount of phase current variation when the AD conversion is changed by 1LSB [A] × 2^{11} This bit is used for current control (Task 5).

4.3.8.2. *[VExIQREF]* (q-axis Current Reference Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	IQREF[15:0]	0x0000	R/W	Sets the reference value of q-axis current (16-bit fixed-point data: -1.0 to 1.0, 15 fractional bits). "0x8000" to "0x7FFF": q-axis current reference [A] / Max_I × 2^{15} Max_I: The amount of phase current variation when the AD conversion is changed by 1LSB [A] × 2^{11} This bit is used for current control (Task 5).

4.3.9. d-axis/q-axis Voltage Setting Registers

4.3.9.1. *[VExVD]* (d-axis Voltage Register)

Bit	Bit symbol	After reset	Type	Function
31:0	VD[31:0]	0x00000000	R/W	Sets the value of d-axis voltage (32-bit fixed-point data: -1.0 to 1.0, 31 fractional bits). "0x80000000" to "0x7FFFFFFF": d-axis voltage[V] / Max_V × 2^{31} Max_V: The amount of supply voltage variation when the AD conversion is changed by 1LSB [V] × 2^{12} This bit is updated when current control (Task 5) is executed. This bit is used for output coordinate axis transformation (Task 7). This bit is used for input coordinate axis transformation (Task 4) in motor voltage equation.

4.3.9.2. *[VExVQ]* (q-axis Voltage Register)

Bit	Bit symbol	After reset	Type	Function
31:0	VQ[31:0]	0x00000000	R/W	Sets the value of q-axis voltage (32-bit fixed-point data: -1.0 to 1.0, 31 fractional bits). "0x80000000" to "0x7FFFFFFF": q-axis voltage[V] / Max_V × 2^{31} Max_V: The amount of supply voltage variation when the AD conversion is changed by 1LSB [V] × 2^{12} This bit is updated when current control (Task 5) is executed. This bit is used for output coordinate axis transformation (Task 7). This bit is used for input coordinate axis transformation (Task 4) in motor voltage equation.

4.3.9.3. *[VExVDQ]* (Voltage Scalar Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VDQ[15:0]	0x0000	R/W	Sets value of voltage scalar for d-axis voltage ***[VExVD]*** and q-axis voltage ***[VExVQ]***, or sets the limitation value of axis for d-axis voltage ***[VExVD]*** and q-axis voltage ***[VExVQ]*** when direction scalar is limited. "0x0000" to "0x7FFF": Voltage [V] / Max_V × 2^{15} Max_V: The amount of supply voltage variation when the AD conversion is changed by 1LSB [V] × 2^{12} This bit is updated for current control (Task 5) when expansion control is enabled.

4.3.9.4. *[VExVSLIM]* (Voltage Scalar Limitation Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VSLIM[15:0]	0x0000	R/W	Sets the limitation value of voltage scalar for d-axis voltage ***[VExVD]*** and q-axis voltage ***[VExVQ]***. "0x0000" to "0x7FFF": Voltage [V] / Max_V × 2^{15} Max_V: The amount of supply voltage variation when the AD conversion is changed by 1LSB [V] × 2^{12} Limitation of scalar voltage is not effective when "0x0000" is set. This bit is used for current control (Task 5) when expansion control is enabled.

4.3.10. Coefficient Registers for PI Control

4.3.10.1. *[VExCIDKI]* (d-axis Current Integral Coefficient Setting Register for PI Control)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	CIDKI[15:0]	0x0000	R/W	Sets the PI control integral coefficient for d-axis. "0x8000" to "0x7FFF" Used for current control (Task 5).

4.3.10.2. *[VExCIDKP]* (d-axis Current Proportional Coefficient Setting Register for PI Control)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	CIDKP[15:0]	0x0000	R/W	Sets the PI control proportional coefficient for d-axis. "0x8000" to "0x7FFF" Used for current control (Task 5).

4.3.10.3. *[VExCIDKG]* (d-axis Coefficient Range Setting Register for PI Control)

Bit	Bit symbol	After reset	Type	Function
31:8	-	0	R	Read as "0".
7:4	CIDKPG[3:0]	0000	R/W	Selects a PI control d-axis proportional coefficient range. 0000: 1 / 1 0001: 1 / 2^4 0010: 1 / 2^8 0011: 1 / 2^{12} 0100: 1 / 2^{16} 0101 to 1000: Reserved 1001: 2 1010: 2^2 1011: 2^3 1100: 2^4 1101 to 1111: Reserved This bit is used for current control (Task 5).

Bit	Bit symbol	After reset	Type	Function
3:0	CIDKIG[3:0]	0000	R/W	Selects a PI control d-axis integral coefficient range. 0000: 1 / 1 0001: 1 / 2^4 0010: 1 / 2^8 0011: 1 / 2^{12} 0100: 1 / 2^{16} 0101 to 1111: Reserved This bit is used for current control (Task 5).

4.3.10.4. *[VExCIQKI]* (q-axis Current Integral Coefficient Setting Register for PI Control)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	CIQKI[15:0]	0x0000	R/W	Sets the PI control integral coefficient for q-axis. "0x8000" to "0x7FFF" Used for current control (Task 5).

4.3.10.5. *[VExCIQKP]* (q-axis Current Proportional Coefficient Setting Register for PI Control)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	CIQKP[15:0]	0x0000	R/W	Sets the PI control proportional coefficient for q-axis. "0x8000" to "0x7FFF" Used for current control (Task 5).

4.3.10.6. *[VExCIQKG]* (q-axis Coefficient Range Setting Register for PI Control)

Bit	Bit symbol	After reset	Type	Function
31:8	-	0	R	Read as "0".
7:4	CIQKPG[3:0]	0000	R/W	Selects a PI control q-axis proportional coefficient range. 0000: 1 / 1 0001: 1 / 2^4 0010: 1 / 2^8 0011: 1 / 2^{12} 0100: 1 / 2^{16} 0101 to 1000: Reserved 1001: 2 1010: 2^2 1011: 2^3 1100: 2^4 1101 to 1111: Reserved This bit is used for current control (Task 5).
3:0	CIQKIG[3:0]	0000	R/W	Selects a PI control q-axis integral coefficient range. 0000: 1 / 1 0001: 1 / 2^4 0010: 1 / 2^8 0011: 1 / 2^{12} 0100: 1 / 2^{16} 0101 to 1111: Reserved This bit is used for current control (Task 5).

4.3.10.7. *[VExVDIH]* (d-axis Voltage Integral Coefficient Store Register for PI Control (Upper 32 bits))

Bit	Bit symbol	After reset	Type	Function
31:0	VDIH[31:0]	0x00000000	R/W	Stores the upper 32 bits of the integral component (VDI) for d-axis on PI control.

4.3.10.8. *[VExVDILH]* (d-axis Voltage Integral Coefficient Store Register for PI Control (Lower 16 bits))

Bit	Bit symbol	After reset	Type	Function
31:16	VDILH[15:0]	0x0000	R/W	Stores the lower 16 bits of the integral component (VDI) for d-axis on PI control.
15:0	-	0	R	Read as "0".

Note: The VDI data is 64-bit fixed-point data (-1.0 to 1.0, 63 fractional bits).

Note: The VDI data consists of 48 bits.

4.3.10.9. *[VExVQIH]* (q-axis Voltage Integral Coefficient Store Register for PI Control (Upper 32 bits))

Bit	Bit symbol	After reset	Type	Function
31:0	VQIH[31:0]	0x00000000	R/W	Stores the upper 32 bits of the integral component (VQI) for q-axis on PI control. Used for current control (Task 5).

4.3.10.10. *[VExVQILH]* (q-axis Voltage Integral Coefficient Store Register for PI Control (Lower 16 bits))

Bit	Bit symbol	After reset	Type	Function
31:16	VQILH[15:0]	0x0000	R/W	Stores the lower 16 bits of the integral component (VQI) for q-axis on PI control. Used for current control (Task 5).
15:0	-	0	R	Read as "0".

Note: The VQI data is 64-bit fixed-point data (-1.0 to 1.0, 63 fractional bits).

Note: The VQI data consists of 48 bits.

4.3.11. *[VExPIOLIM]* (Output Limitation Register for PI Control)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	PIOLIM[15:0]	0x0000	R/W	PI control output limit value setting (0.0 to 1.0, 15 fixed-point data:). "0x0000" to "0x7FFF" This bit is used for current control (Task 5).

4.3.12. *[VExFPWMCHG]* (PWM Switching Speed Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	FPWMCHG [15:0]	0x0000	R/W	Sets the switching speed when PWM shift is enabled. "0x0000" to "0x7FFF": Switching speed [Hz] / Max_Hz × 2^{15} (Max_Hz: Maximum rotational speed [Hz]) Used in output control 1, 2 (Task 0 and 9) when 1 shunt current is detected and PWM shift 1 is enabled.

4.3.13. *[VExPWMOFS]* (PWM Shift 2 Offset Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	PWMOFS [15:0]	0x0000	R/W	Offset to be removed from duty value when shift is enabled in output control 2 (Task 9). "0x0000" to "0x7FFF"

4.3.14. SIN/COS Registers

4.3.14.1. *[VExCOS]* (Cosine Value Register of θ)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	COS[15:0]	0x0000	R/W	Sets the cosine value on the ***[VExTHETA]*** (16-bit fixed-point data: -1.0 to 1.0, 15 fractional bits). "0x8000" to "0x7FFF" This bit is updated when SIN/COS calculation (Task 6) is executed. This bit is used for output coordinate axis transformation (Task 7).

4.3.14.2. *[VExSIN]* (Sine Value Register of θ)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	SIN[15:0]	0x0000	R/W	Sets the sine value on the ***[VExTHETA]*** (16-bit fixed-point data: -1.0 to 1.0, 15 fractional bits). "0x8000" to "0x7FFF" This bit is updated when SIN/COS calculation (Task 6) is executed. This bit is used for output coordinate axis transformation (Task 7).

4.3.14.3. *[VExCOSM]* (Previous Cosine Value Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	COSM[15:0]	0x0000	R/W	Stores the previous value of the ***[VExCOS]*** register (16-bit fixed-point data: -1.0 to 1.0, 15 fractional bits). "0x8000" to "0x7FFF" This bit is updated when SIN/COS calculation (Task 6) is executed. This bit is used for input coordinate axis transformation (Task 4).

4.3.14.4. *[VExSINM]* (Previous Sine Value Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	SINM[15:0]	0x0000	R/W	Stores the previous value of the ***[VExSIN]*** register. (16-bit fixed-point data: -1.0 to 1.0, 15 fractional bits) "0x8000" to "0x7FFF" This bit is updated when SIN/COS calculation (Task 6) is executed. This bit is used for input coordinate axis transformation (Task 4).

4.3.15. Sector Information Registers

4.3.15.1. *[VExSECTOR]* (Sector Information Register)

Bit	Bit symbol	After reset	Type	Function
31:4	-	0	R	Read as "0".
3:0	SECTOR[3:0]	0x0	R/W	Indicates the sector information. Setting value: "0x0" to "0xB" A rotational position when output is expressed in the sector that is divided into 12 sectors by 30-degree. This bit is updated when output phase transformations (Task 8 and Task 11) are executed. Used in output control 1 (Task 0) and trigger generation (Task 1).

4.3.15.2. *[VExSECTORM]* (Previous Sector Information Register)

Bit	Bit symbol	After reset	Type	Function
31:4	-	0	R	Read as "0".
3:0	SECTORM [3:0]	0x0	R/W	Stores the previous value of the ***[VExSECTOR]*** register. "0x0" to "0xB" Updated when output phase transformation (Tasks 8, 11) and trigger generation (Task 1) are executed. Used in input process (Task 2).

4.3.16. 3-phase Current Registers

4.3.16.1. *[VExIAO]* (a-phase Zero-current Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	IAO[15:0]	0x0000	R/W	Stores the result of AD conversion for a-phase at zero-current. (Stores the result of AD conversion for a-phase current when the motor is at stop.) This bit is updated in the input process 1 (Task 2) when zero-current detection mode is selected. The result of AD conversion for a-phase at zero-current is stored in <IAO[15:4]> when the result of AD conversion is captured. <IAO[3:0]> stores "0". Used for input process 2 (Task 10).

4.3.16.2. *[VExIBO]* (b-phase Zero-current Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	IBO[15:0]	0x0000	R/W	Stores the result of AD conversion for b-phase at zero-current. (Stores the result of AD conversion for b-phase current when the motor is at stop.) This bit is updated in the input process 1 (Task 2) when zero-current detection mode is selected. The result of AD conversion for a-phase at zero-current is stored in <IBO[15:4]> when the result of AD conversion is captured. <IBO[3:0]> stores "0". Used for input process 2 (Task 10).

4.3.16.3. *[VExICO]* (c-phase Zero-current Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	ICO[15:0]	0x0000	R/W	Stores the result of AD conversion for c-phase at zero-current. (Stores the result of AD conversion for c-phase current when the motor is at stop.) This bit is updated in the input process 1 (Task 2) when zero-current detection mode is selected. The result of AD conversion for a-phase at zero-current is stored in <ICO[15:4]> when the result of AD conversion is captured. <ICO[3:0]> stores "0". Used for input process 2 (Task 10).

4.3.16.4. *[VExIAADC]* (ADC Conversion Result Register for a-phase Current)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	IAADC[15:0]	0x0000	R/W	Stores the result of AD conversion for a-phase current ("0x0000" to "0xFFFF"). This bit is updated when input processes (Task 2 and Task 10) are executed. The result of AD conversion for a-phase is stored in <IAADC[15:4]> when the result of AD conversion is captured. <IAADC[3:0]> stores "0".

4.3.16.5. *[VExIBADC]* (ADC Conversion Result Register for b-phase Current)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	IBADC[15:0]	0x0000	R/W	Stores the result of AD conversion for b-phase current ("0x0000" to "0xFFFF"). This bit is updated when input processes (Task 2 and Task 10) are executed. The result of AD conversion for b-phase is stored in <IBADC[15:4]> when the result of AD conversion is captured. <IBADC[3:0]> stores "0".

4.3.16.6. *[VExICADC]* (ADC Conversion Result Register for c-phase Current)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	ICADC[15:0]	0x0000	R/W	Stores the result of AD conversion for c-phase current ("0x0000" to "0xFFFF"). This bit is updated when input processes (Task 2 and Task 10) are executed. The result of AD conversion for c-phase is stored in <ICADC[15:4]> when the result of AD conversion is captured. <ICADC[3:0]> stores "0".

4.3.16.7. *[VExIA]* (a-phase Current Register)

Bit	Bit symbol	After reset	Type	Function
31:0	IA[31:0]	0x00000000	R/W	a-phase current (32-bit fixed point data: -1.0 to 1.0, 31 fractional bits) "0x80000000" to "0x7FFFFFFF" The value is updated when input process (Task 2, Task 10) is executed. The value is used on input phase transformation (Task 3).

4.3.16.8. *[VExIB]* (b-phase Current Register)

Bit	Bit symbol	After reset	Type	Function
31:0	IB[31:0]	0x00000000	R/W	b-phase current (32-bit fixed point data: -1.0 to 1.0, 31 fractional bits) "0x80000000" to "0x7FFFFFFF" The value is updated when input process (Task 2, Task 10) is executed. The value is used on input phase transformation (Task 3).

4.3.16.9. *[VExIC]* (c-phase Current Register)

Bit	Bit symbol	After reset	Type	Function
31:0	IC[31:0]	0x00000000	R/W	c-phase current (32-bit fixed point data: -1.0 to 1.0, 31 fractional bits) "0x80000000" to "0x7FFFFFFF" The value is updated when input process (Task 2, Task 10) is executed. The value is used on input phase transformation (Task 3).

4.3.17. DC Supply Voltage Registers

4.3.17.1. *[VExVDC]* (DC Supply Voltage Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VDC[15:0]	0x0000	R/W	Sets a supply voltage (16-bit fixed-point data: 0 to 1.0, 15 fractional bits). "0x0000" to "0x7FFF" The actual voltage value can be calculated as: the value of VDC × the value of Max_V / 2^{15} (Max_V: The amount of supply voltage variation when AD conversion is changed by 1LSB [V] × 2^{12}) This bit is updated when ***[VExMODE]***<VDCSEL> is set to "0" and input processes (Task 2 and Task 10) are executed. This bit is used for output phase transformation (Tasks 8 and 11).

4.3.17.2. *[VExVDCL]* (DC2 Supply Voltage Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VDCL[15:0]	0x0000	R/W	Sets a supply voltage (16-bit fixed-point data: 0 to 1.0, 15 fractional bits). "0x0000" to "0x7FFF" The actual voltage value can be calculated as: the value of VDCL × the value of Max_V / 2^{15} (Max_V: The amount of supply voltage variation when AD conversion is changed by 1LSB [V] × 2^{12}) This bit is updated when ***[VExMODE]***<VDCSEL> is set to "1" and input processes (Task 2 and Task 10) are executed.

4.3.18. d-axis/q-axis Current Registers

4.3.18.1. *[VExID]* (d-axis Current Register)

Bit	Bit symbol	After reset	Type	Function
31:0	ID[31:0]	0x00000000	R/W	Sets a d-axis current (32-bit fixed-point data: -1.0 to 1.0, 31 fractional bits). "0x80000000" to "0x7FFFFFFF" The actual voltage value can be calculated as: the value of ID × the value of Max_I / 2^{31} (Max_I: The amount of phase current variation when the AD conversion is changed by 1LSB [A] × 2^{11}) This bit is updated when input coordinate axis transformation (Task 4) is executed. This bit is used for current control (Task 5).

4.3.18.2. *[VExIQ]* (q-axis Current Register)

Bit	Bit symbol	After reset	Type	Function
31:0	IQ[31:0]	0x00000000	R/W	Sets a q-axis current (32-bit fixed-point data: -1.0 to 1.0, 31 fractional bits). "0x80000000" to "0x7FFFFFFF" The actual voltage value can be calculated as: the value of IQ × the value of Max_I / 2^{31} (Max_I: The amount of phase current variation when the AD conversion is changed by 1LSB [A] × 2^{11}) This bit is updated when input coordinate axis transformation (Task 4) is executed. This bit is used for current control (Task 5).

4.3.19. *[VExTADC]* (ADC Conversion Time Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	TADC[15:0]	0x0000	R/W	Sets an ADC conversion time (Refer to Figure 4.4). "0x0000" to "0xFFFF": ADC conversion time [s] / PWM clock period [s] Performs forward correction for a trigger timing at the PWM end when PWM shift 1 output (Shift 1 is enabled and a low-speed flag is "1") is used in 1-shunt current detection mode. Used for trigger generation (Task 1).

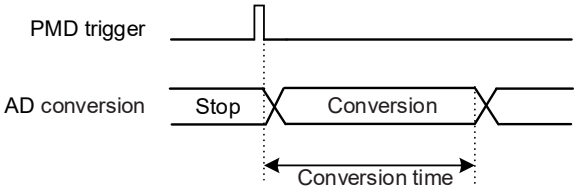


Figure 4.4 ADC Conversion Time

4.3.20. 3-phase PWM Duty Registers

4.3.20.1. *[VExCMPU]* (U-phase PWM Duty Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VCMPU[15:0]	0x0000	R/W	Sets the U-phase PWM duty. "0x0000" to "0x8000" This bit is updated when output controls (Task 0 and Task 9) are executed. This bit is used for trigger generation (Task 1).

Note: Setting PMD's mode select register to VE mode can control U-phase PWM duty of PMD with this register. For details, refer to "Advanced Programmable Motor Control Circuit 2" in the reference manual.

4.3.20.2. *[VExCMPV]* (V-phase PWM Duty Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VCMPV[15:0]	0x0000	R/W	Sets the V-phase PWM duty. "0x0000" to "0x8000" This bit is updated when output controls (Task 0 and Task 9) are executed. This bit is used for trigger generation (Task 1).

Note: Setting PMD's mode select register to VE mode can control V-phase PWM duty of PMD with this register. For details, refer to "Advanced Programmable Motor Control Circuit 2" in the reference manual.

4.3.20.3. *[VExCMPW]* (W-phase PWM Duty Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0."
15:0	VCMPW[15:0]	0x0000	R/W	Sets the W-phase PWM duty. "0x0000" to "0x8000" This bit is updated when output controls (Task 0 and Task 9) are executed. This bit is used for trigger generation (Task 1).

Note: Setting PMD's mode select register to VE mode can control W-phase PWM duty of PMD with this register. For details, refer to "Advanced Programmable Motor Control Circuit 2" in the reference manual.

4.3.21. PWM Output Limitation Registers

4.3.21.1. *[VExPWMMAX]* (PWM Upper Limit Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	PWMMAX [15:0]	0x0000	R/W	Sets the upper limit value of PWM duty output. "0x0000" to "0x8000" This bit is used in output controls (Task 0 and Task 9).

4.3.21.2. *[VExPWMMIN]* (PWM Lower Limit Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	PWMMIN [15:0]	0x0000	R/W	Sets the lower limit value of PWM duty output. "0x0000" to "0x8000" This bit is used in output controls (Task 0 and Task 9).

4.3.22. *[VExOUTCR]* (PMD Output Control Register)

Bit	Bit symbol	After reset	Type	Function
31:9	-	0	R	Read as "0".
8	WPWM	0	R/W	Select W-phase PWM. 0: On/off output 1: PWM output
7	VPWM	0	R/W	Selects V-phase PWM. 0: On/off output 1: PWM output
6	UPWM	0	R/W	Selects U-phase PWM. 0: On/off output 1: PWM output
5:4	WOC[1:0]	00	R/W	Selects W-phase output control. The decode circuit output is controlled by the combination of this bit, <WOC[1:0]>and <WPWM> (Refer to Table 4.3.). For details, refer to "Conduction Control Circuit" of the reference manual "Advanced Programmable Motor Control Circuit 2".
3:2	VOC[1:0]	00	R/W	Selects V-phase output control. The decode circuit output is controlled by the combination of this bit, <VOC[1:0]>and <VPWM> (Refer to Table 4.2.). For details, refer to "Conduction Control Circuit" of the reference manual "Advanced Programmable Motor Control Circuit 2".
1:0	UOC[1:0]	00	R/W	Selects U-phase output control. The decode circuit output is controlled by the combination of this bit, <UOC[1:0]>and <UPWM> (Refer to Table 4.1.). For details, refer to "Conduction Control Circuit" of the reference manual "Advanced Programmable Motor Control Circuit 2".

Note: Setting PMD's mode select register to VE mode can control ***[PMDxMDOUT]*** of PMD with this register. For details, refer to "Advanced Programmable Motor Control Circuit 2" in the reference manual.

Note: This setting is updated when output controls (Task 0 and Task 9) are executed.

Output control of U, V and W-phase of the PMD is shown below: (This table shows only those combinations that are used in the VE.)

Table 4.1 <UPWM>, <UOC[1:0]> PMD Setting: Output Control of U-phase (UOx, XOx)

Setting		Output	
<UPWM>	<UOC[1:0]>	UOx	XOx
0	00	Off output	Off output
0	01	Off output	On output
1	00	PWMU inverted output	PWMU output
1	11	PWMU output	PWMU inverted output

Table 4.2 <VPWM>, <VOC[1:0]> PMD Setting: Output Control of V-phase (VOx, YOx)

Setting		Output	
<VPWM>	<VOC[1:0]>	VOx	YOx
0	00	Off output	Off output
0	01	Off output	On output
1	00	PWMV inverted output	PWMV output
1	11	PWMV output	PWMV inverted output

Table 4.3 <WPWM>, <WOC[1:0]> PMD Setting: Output Control of W-phase (WOx, ZOx)

Setting		Output	
<WPWM>	<WOC[1:0]>	WOx	ZOx
0	00	Off output	Off output
0	01	Off output	On output
1	00	PWMW inverted output	PWMW output
1	11	PWMW output	PWMW inverted output

4.3.23. Trigger Generation Registers

4.3.23.1. *[VExTRGCRC]* (Synchronous Trigger Correction Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	TRGCRC [15:0]	0x0000	R/W	Corrects the trigger timing. "0x0000" to "0x7FFF": Correction time[s] / PWM period[s] This bit is enabled only when PWM shift is disabled in the 1-shunt current detection or when Shift 1 is enabled. This bit is used to backward correct the trigger timing. This bit is used for trigger generation (Task 1).

4.3.23.2. *[VExTRGCMP0]* (PMD Trigger Timing Setting Register 0)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VTRGCMP0 [15:0]	0x0000	R/W	Specifies the trigger timing for sampling ADC in synchronization with the PMD (PMD setting). "0x0000": Prohibited "0x0001" to "0x7FFF": Trigger timing "0x8000" to "0xFFFF": Prohibited This bit is enabled when one of the following PMD trigger modes is selected: The coincidence in the first half of the PWM cycle; The coincidence in the latter half of the PWM cycle; The coincidence in the first and later halves of the PWM cycle This bit is updated in trigger generation (Task 1) when PWM shift is disabled in the 1-shunt current detection or when Shift 1 is enabled.

Note: Setting PMD's mode select register to VE mode can control the trigger comparison 0 of PMD with this register. For details, refer to "Advanced Programmable Motor Control Circuit 2" in the reference manual.

4.3.23.3. *[VExTRGCMP1]* (PMD Trigger Timing Setting Register 1)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VTRGCMP1 [15:0]	0x0000	R/W	Specifies the trigger timing for sampling ADC in synchronization with the PMD (PMD setting). "0x0000": Prohibited "0x0001" to "0x7FFF": Trigger timing "0x8000" to "0xFFFF": Prohibited This bit is enabled when one of the following PMD trigger modes is selected: The coincidence in the first half of the PWM cycle; The coincidence in the latter half of the PWM cycle; The coincidence in the first and later halves of the PWM cycle This register is invalid when the trigger output mode in PMD is set to the trigger selected output (***[PMDxTRGMD]*** <TRGOUT> = 1). This bit is updated in trigger generation (Task 1) when PWM shift is disabled in the 1-shunt current detection or when Shift 1 is enabled.

Note: Setting PMD's mode select register to VE mode can control the trigger comparison 1 of PMD with this register. For details, refer to "Advanced Programmable Motor Control Circuit 2" in the reference manual.

4.3.23.4. *[VExTRGSEL]* (Synchronous Trigger Selection Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:3	-	0	R	Read as "0".
2:0	VTRGSEL[2:0]	0x0	R/W	Select the synchronous trigger number that is output at the timing specified in ***[VExTRGCMP0]*** (PMD setting). "0x0" to "0x5": Output trigger number "0x6", "0x7": Prohibited This bit is enabled when trigger selection output ***[PMDxTRGMD]***<TRGOUT> = 1 is selected as PMD trigger output mode. This bit is updated to the value of ***[VExSECTOR]*** / 2 in trigger generation (Task 1).

Note: Setting PMD's mode select register to VE mode can control the trigger output selection of PMD with this register. For details, refer to "Advanced Programmable Motor Control Circuit 2" in the reference manual.

4.3.24. *[VExATANX]* (ATAN2 Function X Input Register)

Bit	Bit symbol	After reset	Type	Function
31:0	ATANX [31:0]	0x0000 0000	R/W	X input 32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits) Used for ATAN2 (Task 12).

4.3.25. *[VExATANY]* (ATAN2 Function Y Input Register)

Bit	Bit symbol	After reset	Type	Function
31:0	ATANY [31:0]	0x0000 0000	R/W	Y input 32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits) Used for ATAN2 (Task 12).

4.3.26. *[VExATANTHETA]* (ATAN2 Function Phase Difference Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	ATANTHETA [15:0]	0x0000	R/W	Phase value 16-bit fixed-point data (-1.0 to 1.0, 15 fractional bits) (-180 to 180 degrees) Used for ATAN2 (Task 12).

4.3.27. *[VExSQRTIN]* (SQRT Function Input Register)

Bit	Bit symbol	After reset	Type	Function
31:0	SQRTIN [31:0]	0x0000 0000	R/W	Input 32-bit fixed-point data (0.0 to 4.0, 15 fractional bits) "0x00000000" to "0x0001FFFF" Used for SQRT (Task 13).

4.3.28. *[VExSQRTOUT]* (SQRT Function Output Register)

Bit	Bit symbol	After reset	Type	Function
31:0	SQRTOUT [31:0]	0x0000 0000	R/W	Output 32-bit fixed-point data (0.0 to 4.0, 15 fractional bits) "0x00000000" to "0x0001FFFF" Used for SQRT (Task 13).

4.3.29. *[VExDELTA]* (Declination Angle Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	DELTA[15:0]	0x0000	R/W	Sets the current declination angle on the d-q coordinate. (-180 to 180 degrees) "0x8000" to "0x7FFF": The declination angle: [deg] / 360 × 2^{15} This bit is updated when ATAN calculation is enabled on coordinate axis transformation (Task 4) or when voltage scalar limitation is enabled in current control (Task 5).

4.3.30. d-axis/q-axis Induced Voltage Register

4.3.30.1. *[VExEDDELTA]* (d-axis Induced Voltage (Declination Output) Register)

Bit	Bit symbol	After reset	Type	Function
31:0	EDDELTA [31:0]	0x0000 0000	R/W	d-axis induced voltage (Declination angle output) 32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits) d-axis induced voltage (Declination angle output) [V] / Max_V × 2^{31} Max_V: Amount of change in power supply voltage that AD conversion result changes by 1 LSB [V] × 2^{12} Updated when input coordinate axis transformation (Park transformation) (Task 4) is executed (declination angle calculation).

4.3.30.2. *[VExEQDELTA]* (q-axis Induced Voltage (Declination Output) Register)

Bit	Bit symbol	After reset	Type	Function
31:0	EQDELTA [31:0]	0x0000 0000	R/W	q-axis induced voltage (Declination angle output) 32-bit fixed-point data (-1.0 to 1.0, 31 fractional bits) q-axis induced voltage (Declination angle output) [V] / Max_V × 2^{31} Max_V: Amount of change in power supply voltage that AD conversion result changes by 1 LSB [V] × 2^{12} Updated when input coordinate axis transformation (Park transformation) (Task 4) is executed (declination angle calculation).

4.3.31. Motor Constant Registers

4.3.31.1. *[VExCPHI]* (Motor Interlinkage Magnetic Flux Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	CPHI[15:0]	0x0000	R/W	Sets the back electromotive force constant of the motor [V/rps] (interlinkage flux [Wb/s]) 16-bit fixed-point data (-16 to 16, 11 fractional bits): The value of back electromotive force constant [V/rps] / Max_V × Max_Hz × 2^{11} / [CPHIG setting] Used when non-interference control is enabled in current control (Task 5).

4.3.31.2. *[VExCLD]* (Motor d-axis Inductance Value Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	CLD[15:0]	0x0000	R/W	Sets the d-axis inductance of the motor constant. 16-bit fixed-point data (-16 to 16, 11 fractional bits): The value of inductance [H] × Max_I / Max_V × Max_Hz × 2π × 2^{11} / [CLG setting] This bit is used in coordinate axis transformation (Task 4) when the declination angle of induced voltage is calculated or when Non-interference control is enabled in current control (Task 5).

4.3.31.3. *[VExCLQ]* (Motor q-axis Inductance Value Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	CLQ[15:0]	0x0000	R/W	Sets the q-axis inductance of the motor constant. 16-bit fixed-point data (-16 to 16, 11 fractional bits): The value of inductance [H] × Max_I / Max_V × Max_Hz × 2π × 2^{11} / [CLG setting] This bit is used in coordinate axis transformation (Task 4) when the declination angle of induced voltage is calculated or when Non-interference control is enabled in current control (Task 5).

4.3.31.4. *[VExCR]* (Motor Resistance Value Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	CR[15:0]	0x0000	R/W	Sets the value of resistance of the motor constant. 16-bit fixed-point data (-16 to 16, 11 fractional bits): The value of resistor [Ω] × Max_I / Max_V × 2^{11} / (***[VExCRG]*** setting) This bit is used in coordinate axis transformation (Task 4) when the declination angle of induced voltage is calculated or when Non-interference control is enabled in current control (Task 5).

4.3.31.5. *[VExCPHIG]* (Motor Magnetic Flux Range Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:8	-	0	R	Read as "0".
7:3	-	00000	R/W	Write as "00000".
2:0	CPHIG[2:0]	000	R/W	Selects the range of magnetic flux of the motor constant. 000: 1 / 1 001: 1 / 2^4 010: 1 / 2^8 011: 1 / 2^{12} 100: 1 / 2^{16} 101 to 111: Reserved Used when non-interference control is enabled in current control (Task 5).

4.3.31.6. *[VExCLG]* (Motor Inductance Range Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:8	-	0	R	Read as "0".
7:3	-	00000	R/W	Write as "00000".
2:0	CLG[2:0]	000	R/W	Selects the range of inductance of the motor constant. 000: 1 / 1 001: 1 / 2^4 010: 1 / 2^8 011: 1 / 2^{12} 100: 1 / 2^{16} 101 to 111: Reserved This bit is used in coordinate axis transformation (Task 4) when the declination angle of induced voltage is calculated or when Non-interference control is enabled in current control (Task 5).

4.3.31.7. *[VExCRG]* (Motor Resistance Range Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:8	-	0	R	Read as "0".
7:3	-	00000	R/W	Write as "00000".
2:0	CRG[2:0]	000	R/W	Selects the range of the resistance of the motor constant. 000: 1 / 1 001: 1 / 2^4 010: 1 / 2^8 011: 1 / 2^{12} 100: 1 / 2^{16} 101 to 111: Reserved This bit is used in coordinate axis transformation (Task 4) when the declination angle of induced voltage is calculated or when Non-interference control is enabled in current control (Task 5).

4.3.32. *[VExVDE]* (d-axis Voltage Register for Non-interference Control)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VDE[15:0]	0x0000	R/W	Indicates the value of d-axis calculation of non-interference (16-bit fixed-point data -1.0 to 1.0, 15 fractional bits). "0x8000" to "0x7FFF": Voltage [V] / Max_V × 2^{15} (Max_V: The amount of voltage variation when the AD conversion is changed by 1LSB [V] × 2^{12}) This bit is updated when Non-interference control is enabled in current control (Task 5).

4.3.33. *[VExVQE]* (q-axis Voltage Register for Non-interference Control)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VQE[15:0]	0x0000	R/W	Indicates the value of q-axis calculation of non-interference (16-bit fixed-point data: -1.0 to 1.0, 15 fractional bits). "0x8000" to "0x7FFF": Voltage [V] / Max_V × 2^{15} (Max_V: The amount of voltage variation when the AD conversion is changed by 1LSB [V] × 2^{12}) This bit is updated when Non-interference control is enabled in current control (Task 5).

4.3.34. *[VExDTC]* (Dead Time Compensation Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	DTC[15:0]	0x0000	R/W	Sets the amount of compensation on dead time control. "0x0000" to "0x1FFF": Dead time[s] / PWM period[s] × 2^{15} This bit is used when dead time compensation is enabled in output controls (Task 0 and Task 9).

4.3.35. *[VExHYS]* (Hysteresis Width Register for Current Polarity Determination)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	HYS[15:0]	0x0000	R/W	Sets the current hysteresis width when current polarity is determined (16-bit fixed-point data: -1.0 to 1.0, 15 fractional bits). "0x8000" to "0x7FFF": Hysteresis width [A] / Max_I × 2^{15} (Max_I: The amount of phase current variation when the AD conversion is changed by 1LSB [A] × 2^{11}) This bit is used in input process (Task 2 and Task 10) when current polarity determination is enabled.

4.3.36. *[VExHYS2]* (Hysteresis Width Register 2 for Current Polarity Determination)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	HYS2[15:0]	0x0000	R/W	Sets a current reverse hysteresis width on polarity determination. (16-bit fixed point data: 0 to 1.0, 15 fractional bits) "0x0000" to "0x7FFF": Hysteresis width [A] / Max_I × 2^{15} (Max_I: Phase current variation when A/D conversion data is changed for 1LSB [A] × 2^{11}) The value is used on input process (Task2 and Task 10) when current polarity determination is enabled. Note: This register is invalid when ***[VExHYS]*** ≥ 0.

4.3.37. *[VExDTCS]* (Dead Time Compensation Control/Status Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:11	-	00000	R/W	Write as "00000".
10:8	ICSTS[2:0]	000	R/W	Indicates status of the current ***[VExIC]*** polarity determination, or selects dead time compensation control setting for ***[VExCMPW]***. 000, 010, 100, 110: Polarity determination is undefined. Dead time compensation is not performed. 001, 101: Positive current is determined. Adds the ***[VExDTC]*** value when dead time compensation is performed. 011, 111: Negative current is determined. Subtracts the ***[VExDTC]*** value when dead time compensation is performed. This bit is updated in input process (Task 2 and Task 10) when current polarity determination is enabled. This bit is used in output controls (Task 0 and Task 9) when dead time compensation is enabled.
7	-	0	R/W	Write as "0".
6:4	IBSTS[2:0]	000	R/W	Indicates status of the current ***[VExIB]*** polarity determination, or selects dead time compensation control setting for ***[VExCMPV]***. 000, 010, 100, 110: Polarity determination is undefined. Dead time compensation is not performed. 001, 101: Positive current is determined. Adds the ***[VExDTC]*** value when dead time compensation is performed. 011, 111: Negative current is determined. Subtracts the ***[VExDTC]*** value when dead time compensation is performed. This bit is updated in input process (Task 2 and Task 10) when current polarity determination is enabled. This bit is used in output controls (Task 0 and Task 9) when dead time compensation is enabled.
3	-	0	R/W	Write as "0".

Bit	Bit symbol	After reset	Type	Function
2:0	IASTS[2:0]	000	R/W	Indicates status of the current ***[VExIA]*** polarity determination, or selects dead time compensation control setting for ***[VExCMPU]***. 000, 010, 100, 110: Polarity determination is undefined. Dead time compensation is not performed. 001, 101: Positive current is determined. Adds the ***[VExDTC]*** value when dead time compensation is performed. 011, 111: Negative current is determined. Subtracts the ***[VExDTC]*** value when dead time compensation is performed. This bit is updated in input process (Task 2 and Task 10) when current polarity determination is enabled. This bit is used in output controls (Task 0 and Task 9) when dead time compensation is enabled.

4.3.38. *[VExTHTCLP]* (Clipped Phase Value Setting Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	THTCLP[15:0]	0x0000	R/W	Sets the clipping phase of the angle of ***[VExTHETA]*** when phase interpolation is performed (16-bit fixed-point data: 0.0 to 1.0, 15 fractional bits). "0x0000" to "0xFFFF": Phase [deg] / 360 × 2^{16} This bit is used in SIN/COS calculation (Task 6) when phase interpolation is enabled.

4.3.39. 2-phase Voltage Registers

4.3.39.1. *[VExVALPHA]* (α-axis Voltage Register)

Bit	Bit symbol	After reset	Type	Function
31:0	VALPHA[31:0]	0x00000000	R/W	α-axis voltage (32-bit fixed point data: -1.0 to 1.0, 31 fractional bits) "0x80000000" to "0x7FFFFFFF" To convert an actual voltage value, use the following formula: Register value × Max_V / 2^{31} (Max_V: Phase voltage variation when A/D conversion data is changed for 1LSB [V] × 2^{11}) The value is updated when output coordinate axis transformation (Task 7) is executed. The value is used on output phase transformation (Task 8, Task 11).

4.3.39.2. *[VExVBETA]* (β-axis Voltage Register)

Bit	Bit symbol	After reset	Type	Function
31:0	VBETA[31:0]	0x00000000	R/W	β-axis voltage (32-bit fixed point data: -1.0 to 1.0, 31 fractional bits) "0x80000000" to "0x7FFFFFFF" To convert an actual voltage value, use the following formula: Register value × Max_V / 2^{31} (Max_V: Phase voltage variation when A/D conversion data is changed for 1LSB [V] × 2^{11}) The value is updated when output coordinate axis transformation (Task 7) is executed. The value is used on output phase transformation (Task 8, Task 11).

4.3.40. 3-phase Voltage Duty Registers

4.3.40.1. *[VExVDUTYA]* (a-phase Voltage Duty Register)

Bit	Bit symbol	After reset	Type	Function
31:0	VDUTYA[31:0]	0x00000000	R/W	a-phase voltage duty (32-bit fixed point data: 0.0 to 1.0, 31 fractional bits) "0x00000000" to "0x7FFFFFFF" The value is updated on output phase transformation (Task 8, Task 11) is executed. The value is used on output control (Task 0, Task 9).

4.3.40.2. *[VExVDUTYB]* (b-phase Voltage Duty Register)

Bit	Bit symbol	After reset	Type	Function
31:0	VDUTYB[31:0]	0x00000000	R/W	b-phase voltage duty (32-bit fixed point data: 0.0 to 1.0, 31 fractional bits) "0x00000000" to "0x7FFFFFFF" The value is updated on output phase transformation (Task 8, Task 11) is executed. The value is used on output control (Task 0, Task 9).

4.3.40.3. *[VExVDUTYC]* (c-phase Voltage Duty Register)

Bit	Bit symbol	After reset	Type	Function
31:0	VDUTYC[31:0]	0x00000000	R/W	c-phase voltage duty (32-bit fixed point data: 0.0 to 1.0, 31 fractional bits) "0x00000000" to "0x7FFFFFFF" The value is updated on output phase transformation (Task 8, Task 11) is executed. The value is used on output control (Task 0, Task 9).

4.3.40.4. *[VExDUTYZERO]* (Zero Vector Duty Register)

Bit	Bit symbol	After reset	Type	Function
31:0	DUTYZERO [31:0]	0x0000 0000	R/W	Zero vector duty 32-bit fixed-point data (0.0 to 1.0, 31 fractional bits) "0x00000000" to "0x7FFFFFFF" Used for output control 1 (Task 0) and output phase transformation 1 (Task 8).

4.3.41. 2-phase Current Registers

4.3.41.1. *[VExIALPHA]* (α-axis Current Register)

Bit	Bit symbol	After reset	Type	Function
31:0	IALPHA[31:0]	0x00000000	R/W	α-axis current (32-bit fixed point data: -1.0 to 1.0, 31 fractional bits) "0x80000000" to "0x7FFFFFFF" The value is updated on input phase transformation (Task 3). The value is used on input coordination axis transformation (Task 4).

4.3.41.2. *[VExIBETA]* (β-axis Current Register)

Bit	Bit symbol	After reset	Type	Function
31:0	IBETA[31:0]	0x00000000	R/W	β-axis current (32-bit fixed point data: -1.0 to 1.0, 31 fractional bits) "0x80000000" to "0x7FFFFFFF" The value is updated on input phase transformation (Task 3). The value is used on input coordination axis transformation (Task 4).

4.3.42. *[VExVDELTA]* (Voltage Declination Angle Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VDELTA[15:0]	0x0000	R/W	Sets the voltage declination angle on the d-q coordinate. (0 to 90 degrees) "0x0000" to "0x4000": The declination angle [deg] / 360 × 2^{16} The value is updated when expansion control of current control (Task 5) is enabled and voltage scalar limitation is valid.

4.3.43. *[VExVDCRC]* (d-axis Voltage Correction Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VDCRC[15:0]	0x0000	R/W	d-axis voltage correction (16-bit fixed point data: -1.0 to 1.0, 15 fractional bits) "0x8000" to "0x7FFF" Used for current control (Task 5).

4.3.44. *[VExVQCRC]* (q-axis Voltage Correction Register)

Bit	Bit symbol	After reset	Type	Function
31:16	-	0	R	Read as "0".
15:0	VQCRC[15:0]	0x0000	R/W	q-axis voltage correction (16-bit fixed point data: -1.0 to 1.0, 15 fractional bits) "0x8000" to "0x7FFF" Used for current control (Task 5).

5. Precautions

- The Advanced Vector Engine 2 (A-VE2) is used in combination with the Programmable Motor Driver (PMD) and 12-bit Analog to Digital converter (ADC).

  – In order for the PMD to use the A-VE2 calculation results, the mode selection register (***[PMDxMODESEL]***) of the PMD should be set to the VE mode.

  – In order for the PMD to use the trigger timing value (***[VExTRGCMP0]***, ***[VExTRGCMP1]***) (Note) calculated by the Trigger generation task, Trigger comparison carrier selection of the PMD should be set to the basic carrier (***[PMDxTRGCR]***<CARSEL> = 0).

Note: The trigger timing is calculated when the PWM shift prohibition or PWM shift 1 is set in 1-shunt current detection mode.

  – When the vector engine is used, the PMD trigger conversion programs (Note) of the ADC according to the synchronous trigger from the PMD should be set.

Note: PMD trigger enable, AIN selection, and the result register selection.

- When attempting to stop supplying the clock, make sure to check whether the A-VE2 is stopping. Note that when the MCU enters STOP1/STOP2 mode, make sure to check whether the A-VE2 is stopping as well.

6. Revision History

Table 6.1 Revision History

Revision	Date	Description
1.0	2026-02-20	- First release

RESTRICTIONS ON PRODUCT USE

Toshiba Corporation and its subsidiaries and affiliates are collectively referred to as “TOSHIBA”.
Hardware, software and systems described in this document are collectively referred to as “Product”.

- TOSHIBA reserves the right to make changes to the information in this document and related Product without notice.
- This document and any information herein may not be reproduced without prior written permission from TOSHIBA. Even with TOSHIBA's written permission, reproduction is permissible only if reproduction is without alteration/omission.
- Though TOSHIBA works continually to improve Product's quality and reliability, Product can malfunction or fail. Customers are responsible for complying with safety standards and for providing adequate designs and safeguards for their hardware, software and systems which minimize risk and avoid situations in which a malfunction or failure of Product could cause loss of human life, bodily injury or damage to property, including data loss or corruption. Before customers use the Product, create designs including the Product, or incorporate the Product into their own applications, customers must also refer to and comply with (a) the latest versions of all relevant TOSHIBA information, including without limitation, this document, the specifications, the data sheets and application notes for Product and the precautions and conditions set forth in the "TOSHIBA Semiconductor Reliability Handbook" and (b) the instructions for the application with which the Product will be used with or for. Customers are solely responsible for all aspects of their own product design or applications, including but not limited to (a) determining the appropriateness of the use of this Product in such design or applications; (b) evaluating and determining the applicability of any information contained in this document, or in charts, diagrams, programs, algorithms, sample application circuits, or any other referenced documents; and (c) validating all operating parameters for such designs and applications. **TOSHIBA ASSUMES NO LIABILITY FOR CUSTOMERS' PRODUCT DESIGN OR APPLICATIONS.**
- **PRODUCT IS NEITHER INTENDED NOR WARRANTED FOR USE IN EQUIPMENTS OR SYSTEMS THAT REQUIRE EXTRAORDINARILY HIGH LEVELS OF QUALITY AND/OR RELIABILITY, AND/OR A MALFUNCTION OR FAILURE OF WHICH MAY CAUSE LOSS OF HUMAN LIFE, BODILY INJURY, SERIOUS PROPERTY DAMAGE AND/OR SERIOUS PUBLIC IMPACT ("UNINTENDED USE").** Except for specific applications as expressly stated in this document, Unintended Use includes, without limitation, equipment used in nuclear facilities, equipment used in the aerospace industry, Class 3 medical devices, equipment used for automobiles, and military vehicles and munitions. **IF YOU USE PRODUCT FOR UNINTENDED USE, TOSHIBA ASSUMES NO LIABILITY FOR PRODUCT.** For details, please contact your TOSHIBA sales representative or contact us via our website.
- Do not disassemble, analyze, reverse-engineer, alter, modify, translate or copy Product, whether in whole or in part.
- Product shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable laws or regulations.
- The information contained herein is presented only as guidance for Product use. No responsibility is assumed by TOSHIBA for any infringement of patents or any other intellectual property rights of third parties that may result from the use of Product. No license to any intellectual property right is granted by this document, whether express or implied, by estoppel or otherwise.
- **ABSENT A WRITTEN SIGNED AGREEMENT, EXCEPT AS PROVIDED IN THE RELEVANT TERMS AND CONDITIONS OF SALE FOR PRODUCT, AND TO THE MAXIMUM EXTENT ALLOWABLE BY LAW, TOSHIBA (1) ASSUMES NO LIABILITY WHATSOEVER, INCLUDING WITHOUT LIMITATION, INDIRECT, CONSEQUENTIAL, SPECIAL, OR INCIDENTAL DAMAGES OR LOSS, INCLUDING WITHOUT LIMITATION, LOSS OF PROFITS, LOSS OF OPPORTUNITIES, BUSINESS INTERRUPTION AND LOSS OF DATA, AND (2) DISCLAIMS ANY AND ALL EXPRESS OR IMPLIED WARRANTIES AND CONDITIONS RELATED TO SALE, USE OF PRODUCT, OR INFORMATION, INCLUDING WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, ACCURACY OF INFORMATION, OR NONINFRINGEMENT.**
- Do not use or otherwise make available Product or related software or technology for any military purposes, including without limitation, for the design, development, use, stockpiling or manufacturing of nuclear, chemical, or biological weapons or missile technology products (mass destruction weapons). Product and related software and technology may be controlled under the applicable export laws and regulations including, without limitation, the Japanese Foreign Exchange and Foreign Trade Law and the U.S. Export Administration Regulations. Export and re-export of Product or related software or technology are strictly prohibited except in compliance with all applicable export laws and regulations.
- Please contact your TOSHIBA sales representative for details as to environmental matters such as the RoHS compatibility of Product. Please use Product in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. **TOSHIBA ASSUMES NO LIABILITY FOR DAMAGES OR LOSSES OCCURRING AS A RESULT OF NONCOMPLIANCE WITH APPLICABLE LAWS AND REGULATIONS.**