

To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: <http://www.renesas.com>

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (<http://www.renesas.com>)

Send any inquiries to <http://www.renesas.com/inquiry>.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: “Standard”, “High Quality”, and “Specific”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as “Specific” without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as “Specific” or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is “Standard” unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
 - “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
 - “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
 - “Specific”: Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

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

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

Application Note

V850E/IA1TM, V850E/IA2TM

32-Bit Single-Chip Microcontrollers

AC Motor Inverter Control Using Vector Operation

V850E/IA1:

μPD703116
μPD703116(A)
μPD703116(A1)
μPD70F3116
μPD70F3116(A)
μPD70F3116(A1)

V850E/IA2:

μPD703114
μPD70F3114

[MEMO]

NOTES FOR CMOS DEVICES

① PRECAUTION AGAINST ESD FOR SEMICONDUCTORS

Note:

Strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it once, when it has occurred. Environmental control must be adequate. When it is dry, humidifier should be used. It is recommended to avoid using insulators that easily build static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work bench and floor should be grounded. The operator should be grounded using wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with semiconductor devices on it.

② HANDLING OF UNUSED INPUT PINS FOR CMOS

Note:

No connection for CMOS device inputs can be cause of malfunction. If no connection is provided to the input pins, it is possible that an internal input level may be generated due to noise, etc., hence causing malfunction. CMOS devices behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using a pull-up or pull-down circuitry. Each unused pin should be connected to V_{DD} or GND with a resistor, if it is considered to have a possibility of being an output pin. All handling related to the unused pins must be judged device by device and related specifications governing the devices.

③ STATUS BEFORE INITIALIZATION OF MOS DEVICES

Note:

Power-on does not necessarily define initial status of MOS device. Production process of MOS does not define the initial operation status of the device. Immediately after the power source is turned ON, the devices with reset function have not yet been initialized. Hence, power-on does not guarantee out-pin levels, I/O settings or contents of registers. Device is not initialized until the reset signal is received. Reset operation must be executed immediately after power-on for devices having reset function.

V850 Series, V850E/IA1, and V850E/IA2 are trademarks of NEC Corporation.

Windows is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

The export of this product from Japan is regulated by the Japanese government. To export this product may be prohibited without governmental license, the need for which must be judged by the customer. The export or re-export of this product from a country other than Japan may also be prohibited without a license from that country. Please call an NEC sales representative.

License not needed: μPD70F3114, 70F3116, 70F3116(A), 70F3116(A1)

The customer must judge

the need for license: μPD703114, 703116, 703116(A), 703116(A1)

• **The information in this document is current as of March, 2002. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC's data sheets or data books, etc., for the most up-to-date specifications of NEC semiconductor products. Not all products and/or types are available in every country. Please check with an NEC sales representative for availability and additional information.**

• No part of this document may be copied or reproduced in any form or by any means without prior written consent of NEC. NEC assumes no responsibility for any errors that may appear in this document.

• NEC does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from the use of NEC semiconductor products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC or others.

• Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of customer's equipment shall be done under the full responsibility of customer. NEC assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.

• While NEC endeavours to enhance the quality, reliability and safety of NEC semiconductor products, customers agree and acknowledge that the possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC semiconductor products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment, and anti-failure features.

• NEC semiconductor products are classified into the following three quality grades:

"Standard", "Special" and "Specific". The "Specific" quality grade applies only to semiconductor products developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of a semiconductor product depend on its quality grade, as indicated below. Customers must check the quality grade of each semiconductor product before using it in a particular application.

"Standard": Computers, office equipment, communications equipment, test and measurement equipment, audio and visual equipment, home electronic appliances, machine tools, personal electronic equipment and industrial robots

"Special": Transportation equipment (automobiles, trains, ships, etc.), traffic control systems, anti-disaster systems, anti-crime systems, safety equipment and medical equipment (not specifically designed for life support)

"Specific": Aircraft, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems and medical equipment for life support, etc.

The quality grade of NEC semiconductor products is "Standard" unless otherwise expressly specified in NEC's data sheets or data books, etc. If customers wish to use NEC semiconductor products in applications not intended by NEC, they must contact an NEC sales representative in advance to determine NEC's willingness to support a given application.

(Note)

(1) "NEC" as used in this statement means NEC Corporation and also includes its majority-owned subsidiaries.

(2) "NEC semiconductor products" means any semiconductor product developed or manufactured by or for NEC (as defined above).

M8E 00.4

Regional Information

Some information contained in this document may vary from country to country. Before using any NEC product in your application, please contact the NEC office in your country to obtain a list of authorized representatives and distributors. They will verify:

- Device availability
- Ordering information
- Product release schedule
- Availability of related technical literature
- Development environment specifications (for example, specifications for third-party tools and components, host computers, power plugs, AC supply voltages, and so forth)
- Network requirements

In addition, trademarks, registered trademarks, export restrictions, and other legal issues may also vary from country to country.

NEC Electronics Inc. (U.S.)

Santa Clara, California
Tel: 408-588-6000
800-366-9782
Fax: 408-588-6130
800-729-9288

NEC do Brasil S.A.

Electron Devices Division
Guarulhos-SP, Brasil
Tel: 11-6462-6810
Fax: 11-6462-6829

NEC Electronics (Europe) GmbH

Duesseldorf, Germany
Tel: 0211-65 03 01
Fax: 0211-65 03 327

• Sucursal en España

Madrid, Spain
Tel: 091-504 27 87
Fax: 091-504 28 60

• Succursale Française

Vélizy-Villacoublay, France
Tel: 01-30-67 58 00
Fax: 01-30-67 58 99

• Filiale Italiana

Milano, Italy
Tel: 02-66 75 41
Fax: 02-66 75 42 99

• Branch The Netherlands

Eindhoven, The Netherlands
Tel: 040-244 58 45
Fax: 040-244 45 80

• Branch Sweden

Taeby, Sweden
Tel: 08-63 80 820
Fax: 08-63 80 388

• United Kingdom Branch

Milton Keynes, UK
Tel: 01908-691-133
Fax: 01908-670-290

NEC Electronics Hong Kong Ltd.

Hong Kong
Tel: 2886-9318
Fax: 2886-9022/9044

NEC Electronics Hong Kong Ltd.

Seoul Branch
Seoul, Korea
Tel: 02-528-0303
Fax: 02-528-4411

NEC Electronics Shanghai, Ltd.

Shanghai, P.R. China
Tel: 021-6841-1138
Fax: 021-6841-1137

NEC Electronics Taiwan Ltd.

Taipei, Taiwan
Tel: 02-2719-2377
Fax: 02-2719-5951

NEC Electronics Singapore Pte. Ltd.

Novena Square, Singapore
Tel: 253-8311
Fax: 250-3583

Major Revisions in This Edition

Page	Contents
Throughout	<ul style="list-style-type: none"> • For V850E/IA1, the following product has been deleted: μPD703117 • For V850E/IA1, the following products have been added: μPD703116, μPD703116(A), μPD703116(A1), μPD70F3116, μPD70F3116(A), μPD70F3116(A1) • The following products (V850E/IA2) have been added: μPD703114, μPD70F3114 • The status of the following product has changed from under development to development complete: μPD70F3116 • Bits defined as reserved words in the device file have been specified (bits whose bit numbers are in angle brackets < >).
p. 15	Addition of Table 1-1 Differences Between V850E/IA1 and V850E/IA2 .
p. 18	Addition of 1.2.2 Pin configuration (top view) .
p. 19	Addition of 1.2.3 Internal block diagram .
p. 72	Addition of cautions to Figure 5-13 Timer Unit Control Register 00 (TUC00) .
p. 73	Modification of Figure 5-14 Block Diagram of Timer 10 (TM10) .
p. 77	Modification of setting values for the PRM02 register in 5.2.3 (1) Timer 1/timer 2 clock selection register (PRM02) settings .
p. 82	Modification of description on Figure 5-20 Signal Edge Selection Register 10 (SESA 10) .
p. 83	Addition of cautions to Figure 5-21 Timer Control Register 10 (TMC10) .
p. 88	Modification of values in Table 6-2 List of Constants .

The mark ★ shows major revised points.

INTRODUCTION

Target Readers

This application note is intended for users who understand the functions of the V850E/IA1, V850E/IA2 and who design application systems that use these microcontrollers. The applicable products are shown below.

- V850E/IA1
Standard products: μ PD703116, 70F3116
Special products: μ PD703116(A), 703116(A1), 70F3116(A), 70F3116(A1)
- V850E/IA2: μ PD703114, 70F3114

Purpose

The purpose of this application note is help users understand the use and composition of the V850E/IA1, V850E/IA2 timer/counter functions (real-time pulse unit). The system example presented here is a 3-phase servo motor control application circuit which features vector operation based on PWM output, encoder input, and A/D converter input.

Organization

This application note is divided into the following sections.

- Introduction
- Functions in application circuit example
- Hardware configuration
- Control system
- Functions of V850E/IA1, V850E/IA2
- Program configuration
- Flow chart
- Program list

How to Use This Manual

It is assumed that the reader of this application note has general knowledge in the fields of electrical engineering, logic circuits, and microcontrollers.

Cautions 1. Application examples in this manual are intended for the “standard” quality models for general-purpose electronic systems. When using an example in this manual for an application that requires the “special” quality grade, evaluate each component and circuit to be actually used to see if they satisfy the required quality standard.

2. To use this manual for special-grade products read the part numbers as follows:

μ PD703116 → μ PD703116(A), 703116(A1)
 μ PD70F3116 → μ PD70F3116(A), 70F3116(A1)

For details of hardware functions (especially register functions, setting methods, etc.)
→ See the **V850E/IA1 Hardware User's Manual, V850E/IA2 Hardware User's Manual.**

For details of instruction functions
→ See the **V850E1 Architecture User's Manual.**

When register format diagrams show values of "0" or "1" in each register, do not set values other than "0" or "1" to those registers.

Bit numbers in the register format drawing for each of the registers that are enclosed in angle brackets < > are defined as reserved words in the device file.

Conventions

Data significance:	Higher digits on the left and lower digits on the right
Active low representation:	$\overline{\text{xxx}}$ (overscore over pin or signal name)
Memory map address:	Higher addresses on the top and lower addresses on the bottom
Note:	Footnote for item marked with Note in the text
Caution:	Information requiring particular attention
Remark:	Supplementary information
Numeric representation:	Binary ... xxxx or xxxxB Decimal ... xxxx Hexadecimal ... xxxxH
Prefix indicating the power of 2 (address space, memory capacity):	K (kilo): $2^{10} = 1,024$ M (mega): $2^{20} = 1,024^2$ G (giga): $2^{30} = 1,024^3$
Data Type:	Word: 32 bits Halfword: 16 bits Byte: 8 bits

Related documents

The related documents indicated in this publication may include preliminary versions. However, preliminary versions are not marked as such.

Documents related to V850E/IA1

Document Name	Document No.
V850E1 Architecture User's Manual	U14559E
μ PD70F3116, 70F3116(A), 70F3116(A1) Data Sheet	U15299E
V850E/IA1 Hardware User's Manual	U14492E
μ PD703114, 70F3114 Data Sheet	To be prepared
V850E/IA2 Hardware User's Manual	U15195E
V850E/IA1, V850E/IA2 AC Motor Inverter Control Using Vector Operation Application Note	This manual

Documents related to development tools (User's Manuals)

Document Name	Document No.	
IE-V850-MC, IE-V850E-MC-A (In-Circuit Emulator)	U14487E	
IE-703116-MC-EM1 (In-Circuit Emulator Option Board for V850E/IA1)	U14700E	
IE-703114-MC-EM1 (In-Circuit Emulator Option Board for V850E/IA2)	To be prepared	
CA850 (Ver. 2.30 or Later) (C Compiler Package)	Operation	U14568E
	C Language	U14566E
	Project Manager	U14569E
	Assembly Language	U14567E
CA850 (Ver. 2.40) (C Compiler Package)	Operation	U15024E
	C Language	U15025E
	Project Manager	U15026E
	Assembly Language	U15027E
ID850 (Ver. 2.40) (Integrated Debugger)	Operation Windows™ Based	U15181E
SM850 (Ver. 2.40) (System Simulator)	Operation Windows Based	U15182E
SM850 (Ver. 2.00 or Later) (System Simulator)	External Part User Open Interface Specifications	U14873E
RX850 (Ver. 3.13 or Later) (Real-Time OS)	Basics	U13430E
	Installation	U13410E
	Technical	U13431E
RX850 Pro (Ver. 3.13) (Real-Time OS)	Fundamental	U13773E
	Installation	U13774E
	Technical	U13772E
RD850 (Ver. 3.01) (Task Debugger)		U13737E
RD850 Pro (Ver. 3.01) (Task Debugger)		U13916E
AZ850 (Ver. 3.0) (System Performance Analyzer)		U14410E
PG-FP3 (Flash Memory Programmer)		U13502E
PG-FP4 (Flash Memory Programmer)		U15260E

CONTENTS

	CHAPTER 1 INTRODUCTION.....	15
	1.1 Outline	15
	1.2 V850E/IA1.....	16
	1.2.1 Features	16
★	1.2.2 Pin configuration (top view).....	18
★	1.2.3 Internal block diagram	19
★	1.3 V850E/IA2.....	20
	1.3.1 Features	20
	1.3.2 Pin configuration (top view).....	22
	1.3.3 Internal block diagram	23
	CHAPTER 2 FUNCTIONS IN APPLICATION CIRCUIT EXAMPLE.....	24
	CHAPTER 3 HARDWARE CONFIGURATION.....	25
	3.1 Operation.....	25
	3.2 System Configuration	26
	3.3 CPU Block.....	27
	3.3.1 Memory map.....	27
	3.3.2 Pin assignments.....	29
	3.3.3 Peripheral I/O.....	36
	3.4 Circuit Diagram	40
	CHAPTER 4 CONTROL SYSTEM	43
	4.1 Overview	43
	4.1.1 Control principles	43
	4.1.2 Control block.....	46
	4.1.3 Motor specifications.....	47
	4.2 Position Control	48
	4.3 Speed Control.....	48
	4.4 Current Control	49
	4.5 Three-Phase Voltage Conversion	49
	4.6 PWM Conversion.....	50
	4.7 Encoder Input Processing	50
	CHAPTER 5 FUNCTIONS OF V850E/IA1 AND V850E/IA2	51
	5.1 PWM Timer Function (Timer 00 [TM00]).....	51
	5.1.1 General.....	52
	5.1.2 Use of PWM timer in application circuit example.....	55
	5.1.3 Register settings	61
	5.2 Encoder Counter Functions (Timer 10 [TM10]).....	73
	5.2.1 General.....	73
	5.2.2 Use of encoder counter in application circuit example.....	76
	5.2.3 Register settings	77
	CHAPTER 6 PROGRAM CONFIGURATION	84

6.1	Program Structure.....	84
6.2	Common Areas.....	86
6.3	Constant Definition	88
6.4	Motor Control Constants.....	90
CHAPTER 7 FLOW CHARTS.....		91
7.1	Main Routine.....	91
7.2	Motor Control Interrupt Servicing (0.4 ms Interval)	97
7.3	Wait Timer Interrupt Servicing (10 ms Interval)	103
7.4	Peripheral I/O Initialization Processing	104
7.5	Common Area Initialization Processing	105
7.6	LED Display Output Processing	106
7.7	sin Calculation Processing	107
7.8	Subroutine Processing for sin Calculation	108
CHAPTER 8 PROGRAM LISTS.....		109
8.1	Definition of Constants	109
8.2	Common Area.....	110
8.3	Main Routine.....	111
8.4	Motor Control Interrupt Servicing (0.4 ms Interval)	113
8.5	Wait Timer Interrupt Servicing (10 ms Interval)	115
8.6	Peripheral I/O Initialization Processing	116
8.7	Common Area Initialization Processing	117
8.8	LED Display Output Processing	117
8.9	Calculation Processing	118
APPENDIX A INDEX		119
★	APPENDIX B REVISION HISTORY	122

LIST OF FIGURES (1/2)

Figure No.	Title	Page
Figure 3-1	LED Display of Minus Sign	26
Figure 3-2	Hardware Configuration Diagram.....	26
Figure 3-3	Memory Map (V850E/IA1)	27
Figure 3-4	Memory Map (V850E/IA2)	28
Figure 3-5	LED Display (V850E/IA1)	37
Figure 3-6	LED Display (V850E/IA2)	37
Figure 3-7	Functions of Port 4 (V850E/IA1)	37
Figure 3-8	Functions of Port 3 (V850E/IA2)	38
Figure 3-9	Circuit Diagram of Application Circuit Example	41
Figure 4-1	Equivalent Circuits	43
Figure 4-2	Control Block.....	46
Figure 5-1	Block Diagram of Timer 00 (TM00) (PWM Mode 0: Symmetric Triangular Wave).....	52
Figure 5-2	Operation Mode Setting.....	56
Figure 5-3	Operation Timing in PWM Mode 0 (Symmetric Triangular Wave).....	59
Figure 5-4	Overall Operation Image of PWM Mode 0 (Symmetric Triangular Wave)	60
Figure 5-5	Timer 0 Clock Selection Register (PRM01).....	61
Figure 5-6	TOMR Write Enable Register 0 (SPEC0).....	61
Figure 5-7	Timer Output Mode Register 0 (TOMR0).....	62
Figure 5-8	PWM Software Timing Output Register 0 (PSTO0).....	64
Figure 5-9	Buffer Registers CM00 to CM03 (BFCM00 to BFCM03)	66
Figure 5-10	Dead-time timer Reload Register 0 (DTRR0).....	67
Figure 5-11	PWM Output Enable Register 0 (POER0).....	68
Figure 5-12	Timer Control Register 00 (TMC00).....	69
Figure 5-13	Timer Unit Control Register 00 (TUC00)	72
Figure 5-14	Block Diagram of Timer 10 (TM10).....	73
Figure 5-15	Mode 4.....	76
Figure 5-16	Timer 1/Timer 2 Clock Selection Register (PRM02)	77
Figure 5-17	Timer 10 Noise Elimination Time Selection Register (NRC10)	78
Figure 5-18	Timer Unit Mode Register 0 (TUM0).....	79
Figure 5-19	Prescaler Mode Register 10 (PRM10)	80
Figure 5-20	Signal Edge Selection Register 10 (SESA10)	81
Figure 5-21	Timer Control Register 10 (TMC10).....	83
Figure 6-1	Program Structure.....	84
Figure 7-1	Main Routine.....	91
Figure 7-2	Motor Control Interrupt Servicing (0.4 ms Interval).....	98
Figure 7-3	Wait Timer Interrupt Servicing (10 ms Interval)	103
Figure 7-4	Peripheral I/O Initialization Processing.....	104
Figure 7-5	Common Area Initialization Processing.....	105

LIST OF FIGURES (2/2)

Figure No.	Title	Page
Figure 7-6	LED Display Output Processing	106
Figure 7-7	sin Calculation Processing	107
Figure 7-8	Subroutine Processing for sin Calculation	108

LIST OF TABLES

Table No.	Title	Page
Table 1-1	Differences Between V850E/IA1 and V850E/IA2	15
Table 3-1	V850E/IA1 Pin Assignment.....	29
Table 3-2	V850E/IA2 Pin Assignment.....	33
Table 3-3	List of Peripheral I/O Functions.....	36
Table 4-1	Motor Specifications	47
Table 5-1	Timer 1 (TM10) Clear Conditions.....	74
Table 6-1	Common Area List.....	86
Table 6-2	List of Constants	88
Table 6-3	List of Motor Control Constants	90

CHAPTER 1 INTRODUCTION

The V850E/IA1 and V850E/IA2 are products in NEC's V850 Series™ of single-chip microcontrollers for real-time control.

1.1 Outline

★ The V850E/IA1 and V850E/IA2 are 32-bit single-chip microcontrollers that realize high-precision inverter control of motors due to high-speed operation. They use the V850E1 CPU of the V850 Series and have on-chip ROM, RAM, a bus interface, DMA controller, a variety of timers including a 3-phase sine-wave PWM timer for motors, serial interfaces, and peripheral functions such as A/D converters. SRAM or ROM can be connected as memory.

The V850E/IA1 has an FCAN (Full Controller Area Network) controller peripheral function.

Table 1-1 lists the differences between the V850E/IA1 and V850E/IA2.

★ **Table 1-1. Differences Between V850E/IA1 and V850E/IA2**

Item		V850E/IA1	V850E/IA2
Maximum operating frequency		50 MHz	40 MHz
Internal ROM	Mask ROM	μPD703116: 256 KB	μPD703114: 128 KB
	Flash memory	μPD70F3116: 256 KB	μPD70F3114: 128 KB
Internal RAM		10 KB	6 KB
Timers	Timers 00, 01	Provided	Buffer register, compare register, and compare match interrupt added
	Timers 10, 11	Provided	Timer 10: Provided; Timer 11: Not provided
	Timers 20, 21	Provided	Provided
	Timer 3	Provided	TO3 output buffer off function by INTP4 input added
	Timer 4	Provided	Provided
Serial interfaces	UART0	Provided	Provided
	UART1	Provided	Provided (pins shared with CSI1)
	UART2	Provided	Not provided
	CSI0	Provided	Provided
	CSI1	Provided	Provided (pins shared with UART1)
	FCAN	Provided	Not provided
Debug support function	NBD	Provided	Not provided
A/D converter	Analog input	Total of two circuits: 16 ch A/D converter 0: 8 ch A/D converter 1: 8 ch	Total of two circuits: 14 ch A/D converter 0: 6 ch A/D converter 1: 8 ch
	AV _{DD} , AV _{REF} pins	Independent pins	Alternate-function pins
Supply voltage		V _{DD3} = 3.3 V ± 0.3 V V _{DD5} = 5.0 V ± 0.5 V	V _{DD} = R _{VDD} = 5.0 V ± 0.5 V Internal regulator
Package		144-pin plastic LQFP	100-pin plastic LQFP

Remark For details, refer to the hardware user's manual of each product.

1.2 V850E/IA1

1.2.1 Features

- Number of instructions 83
- Minimum instruction execution time 20 ns (@ internal 50 MHz operation)
- General-purpose registers 32 bits × 32 registers
- Instruction set V850E1 CPU
 - Signed multiplication (32 bits × 32 bits → 64 bits): 1 or 2 clocks
 - Saturated operation instructions (with overflow/underflow detection function)
 - 32-bit shift instruction: 1 clock
 - Bit manipulation instructions
 - Long/short format load/store instructions
 - Signed load instructions
- Memory space 256 MB linear address space (shared by program and data)
 - Chip select output function: 8 spaces
 - Memory block division function: 2, 4, or 8 MB/block
 - Programmable wait function
 - Idle state insertion function
- External bus interface 16-bit data bus (address/data multiplexed)
 - 16-/8-bit bus sizing function
 - Bus hold function
 - External wait function
- On-chip memory

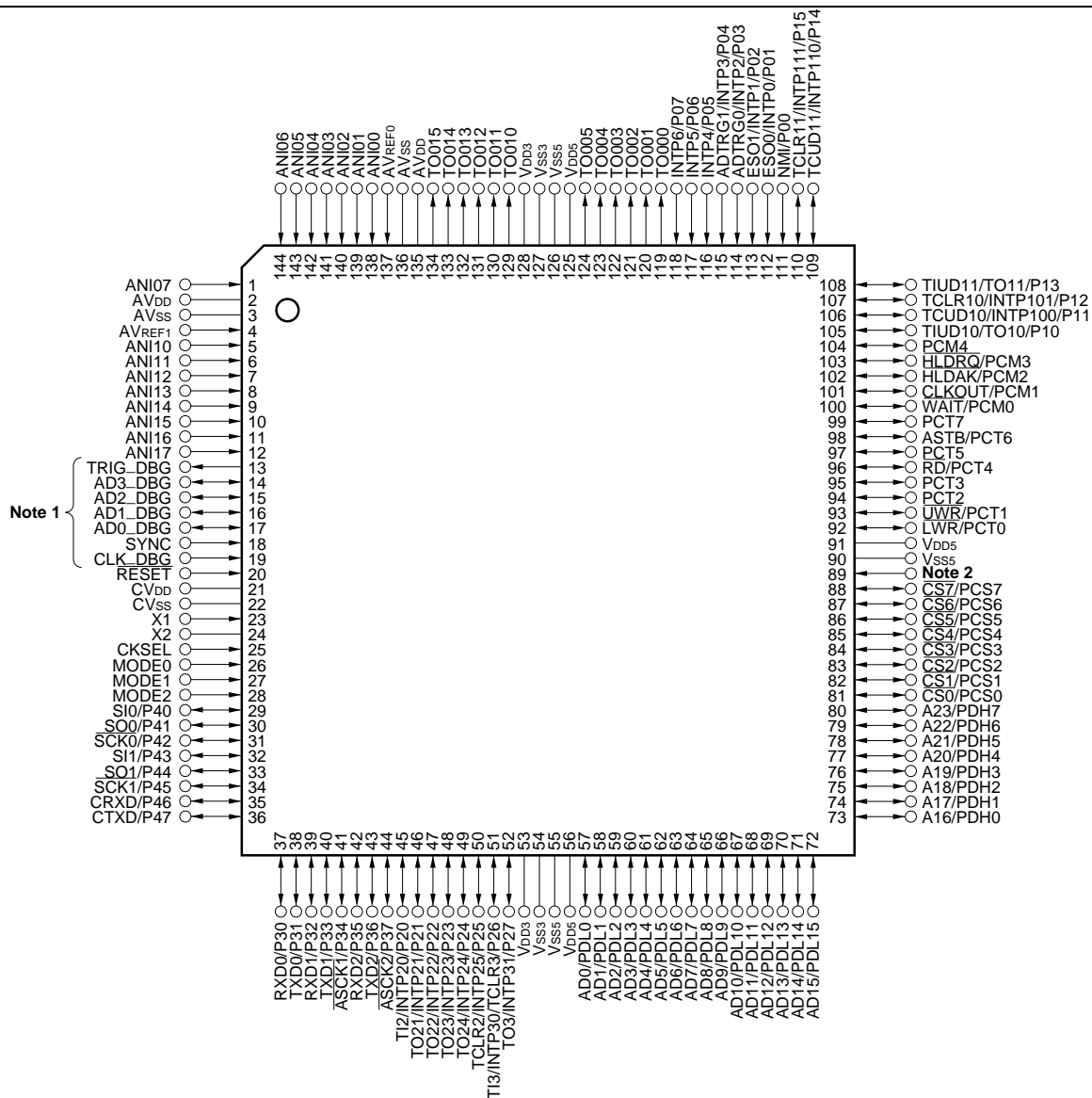
Product Name	Internal ROM	Internal RAM
μPD703116	256 KB (mask ROM)	10 KB
μPD70F3116	256 KB (flash memory)	10 KB
- Interrupts/exceptions External interrupts: 20 (including NMI)
 - Internal interrupts: 45 sources
 - Exceptions: 1 source
 - 8 levels of priority can be specified
- Memory access control SRAM controller
- DMA controller 4-channel configuration
 - Transfer unit: 8 bits/16 bits
 - Maximum transfer count: 65,536 (2¹⁶)
 - Transfer type: 2-cycle transfer
 - Transfer modes: Single transfer, single-step transfer, block transfer
 - Transfer subjects: Memory ↔ Memory, Memory ↔ I/O, I/O ↔ I/O
 - Transfer requests: On-chip peripheral I/O, software
 - Next address setting function
- I/O lines Input ports: 8
I/O ports: 75
- Real-time pulse unit 16-bit timer for 3-phase sine wave PWM inverter control: 2 channels
 - 16-bit up/down counter/timer for 2-phase encoder input: 2 channels
 - General-purpose 16-bit timer/counter: 2 channels
 - General-purpose 16-bit timer/event counter: 1 channel
 - 16-bit interval timer: 1 channel

★

- Serial interface (SIO) Asynchronous serial interface (UART): 3 channels
 Clocked serial interface (CSI): 2 channels
 FCAN (Full Controller Area Network): 1 channel
- NBD (Non Break Debug) function: 1 channel (μ PD70F3116 only)
 - RAM monitoring
 - Event detection
- A/D converter 10-bit resolution A/D converter: 8 channels \times 2 units
- Clock generator Multiplication function ($\times 1$, $\times 2.5$, $\times 5$, $\times 10$) using PLL clock synthesizer
 Divide-by-2 function using external clock input
- Power-saving function HALT, IDLE, and software STOP modes
- Power supply voltage Internal units: 3.3 V, A/D converter: 5 V, External pins: 5V
- Package 144-pin plastic LQFP (fine pitch) (20 \times 20)
- CMOS technology Full static circuits

★ 1.2.2 Pin configuration (top view)

- 144-pin plastic LQFP (fine pitch) (20 × 20)
 - μPD703116GJ-xxx-UEN, 703116GJ(A)- xxx-UEN, 703116GJ(A1)- xxx-UEN
 - μPD70F3116GJ-UEN, 70F3116GJ(A)-UEN, 70F3116GJ(A1)-UEN

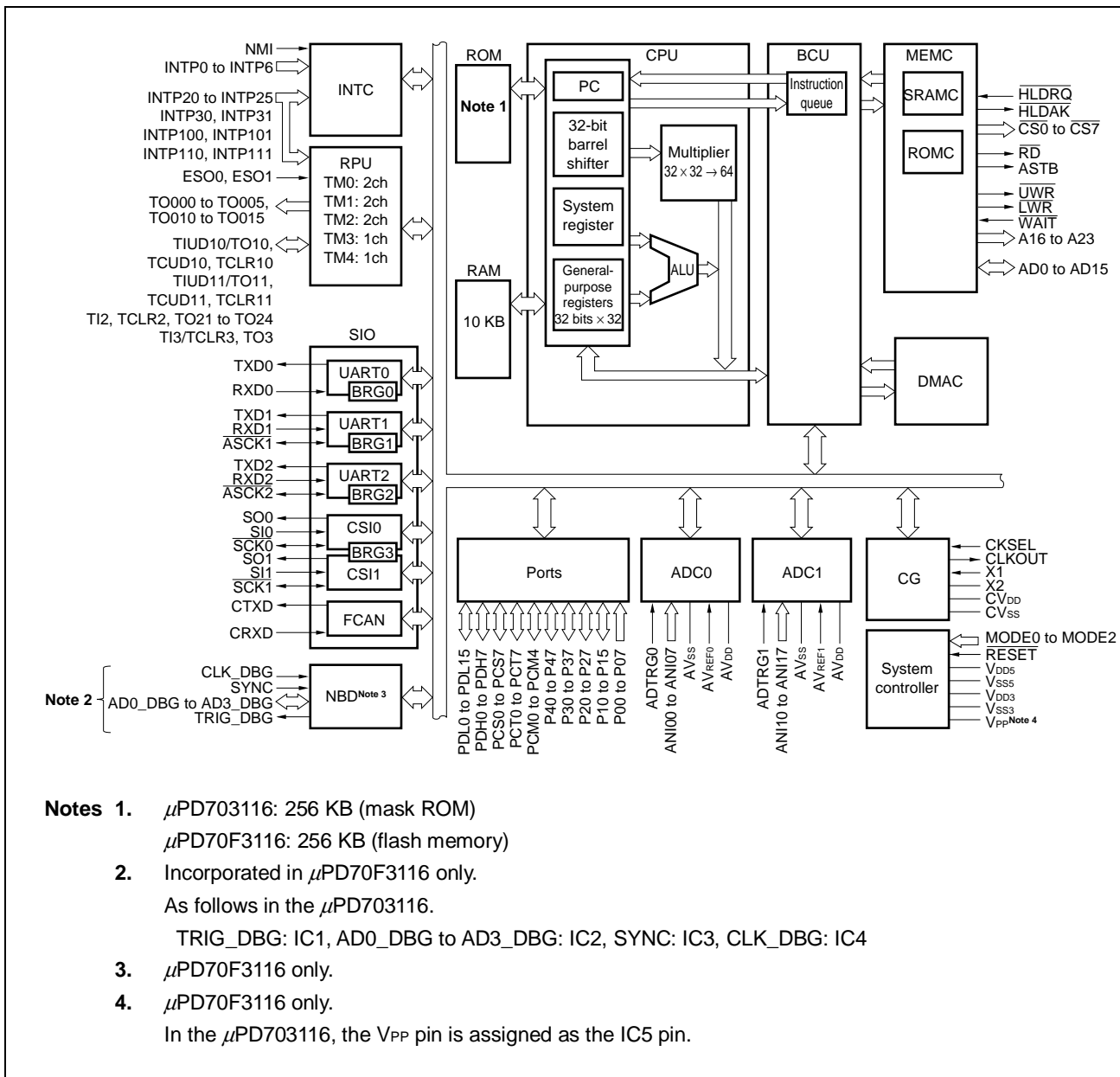


Notes 1. On-chip in μPD70F3116 only.
 As follows in the μPD703116.
 TRIG_DBG: IC1, AD0_DBG to AD3_DBG: IC2, SYNC: IC3, CLK_DBG: IC4

2. μPD703116: IC5
 μPD70F3116: VPP

Cautions 1. When using the μPD70F3116 in normal mode, connect the VPP pin to VSS5.
 2. When using the μPD703116, the processing when pins IC1 to IC5 are not used is as follows:
 Pins IC1 to IC4: Leave open
 Pin IC5: Independently connect to VSS5 via a resistor.

★ 1.2.3 Internal block diagram



★ 1.3 V850E/IA2

1.3.1 Features

- Number of instructions 83
- Minimum instruction execution time
25 ns (@ internal 40 MHz operation)
- General-purpose registers 32 bits × 32 registers
- Instruction set V850E1 CPU
Signed multiplication (32 bits × 32 bits → 64 bits): 1 or 2 clocks
Saturated operation instructions (with overflow/underflow detection function)
32-bit shift instruction: 1 clock
Bit manipulation instructions
Long/short format load/store instructions
Signed load instructions
- Memory space 4 MB linear address space (shared by program and data)
Memory block division function: 2 MB/block
Programmable wait function
Idle state insertion function
- External bus interface 16-bit data bus (address/data multiplexed)
16-/8-bit bus sizing function
External wait function

○ Internal memory

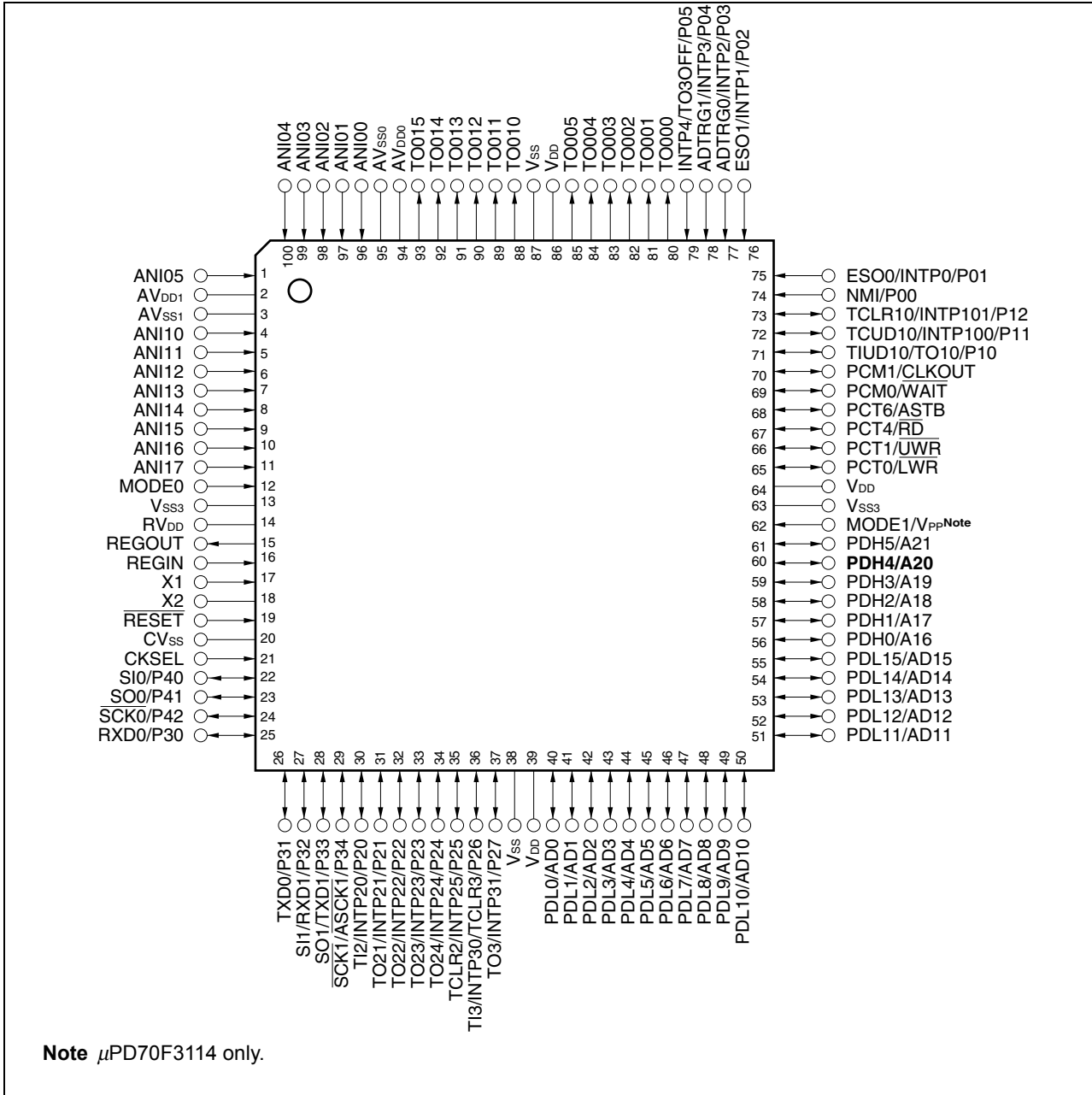
Part Number	Internal ROM	Internal RAM
μPD703114	128 KB (mask ROM)	6 KB
μPD70F3114	128 KB (flash memory)	6 KB

- Interrupts/exceptions External interrupts: 16 (including NMI)
Internal interrupts: 42 sources
Exceptions: 1 source
8 levels of priority can be specified
- Memory access control SRAM controller
- DMA controller 4-channel configuration
Transfer unit: 8 bits/16 bits
Maximum transfer count: 65,536 (2¹⁶)
Transfer type: 2-cycle transfer
Transfer modes: Single transfer, single-step transfer, block transfer
Transfer subjects: Memory ↔ Memory, Memory ↔ I/O, I/O ↔ I/O
Transfer requests: On-chip peripheral I/O, software
Next address setting function
- I/O lines Input ports: 6
I/O ports: 47
- Real-time pulse unit 16-bit timer for 3-phase sine-wave PWM inverter control: 2 channels
16-bit up/down counter/timer for 2-phase encoder input: 1 channel
General-purpose 16-bit timer/counter: 2 channels
General-purpose 16-bit timer/event counter: 1 channel
16-bit interval timer: 1 channel
- Serial interface (SIO) Asynchronous serial interface (UART): 2 channels

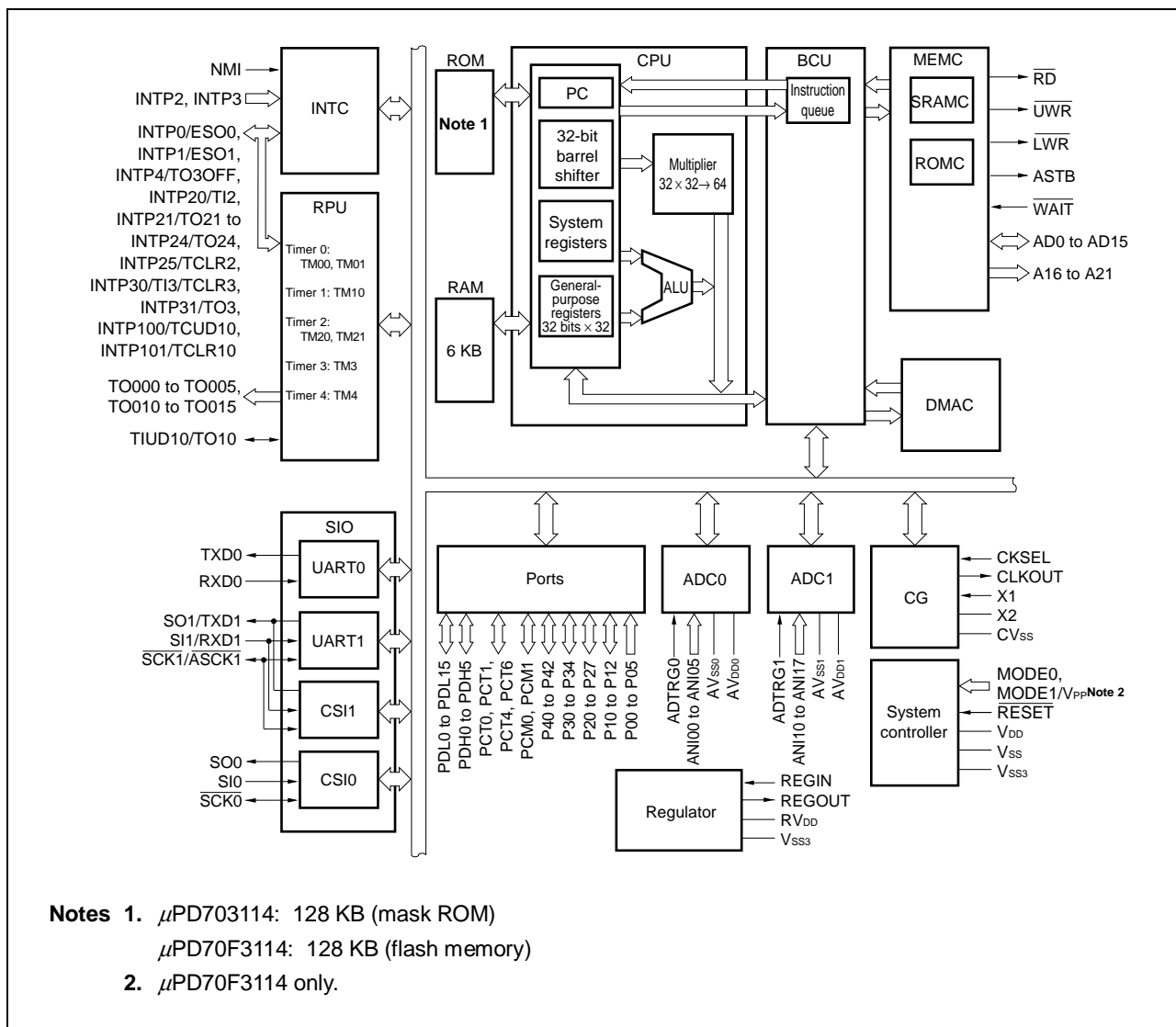
- A/D converter Clocked serial interface (CSI): 2 channels
Of the four channels, two channels are used for both CSI and UART and therefore one or the other function must be selected.
- Regulator 10-bit resolution A/D converter: 6 channels + 8 channels (2 units)
Two power supplies, one for the internal CPU and one for the peripheral interface, are not necessary. A 5 V single-power-supply system can be configured by connecting an N-ch transistor (2SD1950 (VL standard product, surface mount type) or 2SD1581 (independent type) is recommended). If a 3.3 V power supply is available, it can be directly connected to the REGIN pin.
- Clock generator Multiplication function ($\times 1$, $\times 2.5$, $\times 5$, $\times 10$) using PLL clock synthesizer
Divide-by-2 function using external clock input
- Power-saving function HALT, IDLE, and software STOP modes
- Package 100-pin plastic LQFP (fine pitch) (14×14)
- CMOS technology All static circuits

1.3.2 Pin configuration (top view)

- 100-pin plastic LQFP (fine pitch) (14 × 14)
 - μPD703114GC-xxx-8EU
 - μPD70F3114GC-8EU



1.3.3 Internal block diagram



CHAPTER 2 FUNCTIONS IN APPLICATION CIRCUIT EXAMPLE

★ This chapter describes a 3-phase servo motor control application circuit which features vector operation based on PWM output, encoder input, and A/D converter input, as an application example of the timer/counter function (real-time pulse unit) of the V850E/IA1 and V850E/IA2.

The main functions of this application circuit are listed below.

- Performs clockwise rotation, counter-clockwise rotation, and STOP operation.
- Enables rotation speed to be changed using speed volume.
- An array of 16 LEDs displays rotation speeds and positional differences.
- Errors such as overcurrent are monitored and indicated via LED display.

CHAPTER 3 HARDWARE CONFIGURATION

This chapter describes the hardware configuration of the application circuit example.

3.1 Operation

The application circuit's main functions are described below. In this example, when the power is turned on, the application circuit detects the origin position by activating the motor for two rotations. After that, the motor's operation mode is controlled via the operation switches.

(1) Clockwise or counter-clockwise rotation

- The rotor's rotation speed (rpm) varies as indicated by the speed volume indicator.
- The rotation speed ranges from 15 to 1,500 rpm.
- When the rotor is turning, the LED display shows the differential compared to the specified rpm.
- When the operation mode has been changed, the rotor stops turning, waits for 10 ms, then is restarted according to the newly set operation mode (see **Figure 6-1 Program Structure**).

(2) STOP operation

- The rotor is kept at the position where it stops.
- While the rotor is stopped, the LED display shows the differential compared to the stopped position.
- When the mode is changed from STOP mode to clockwise or counter-clockwise rotation, the rotor starts turning.

(3) Errors

- There are three types of errors:
 - Overcurrent error: Error No. 1 (ERR_NO1)
 - Positioning error: Error No. 2 (ERR_NO2)
 - Drive error: Error No. 3 (ERR_NO3)

Caution This application circuit example performs watchdog monitoring which turns off PWM output via hardware when a program loop occurs due to errors other than those listed above.

- When an error has occurred, the current operation mode is stopped and the corresponding error number is displayed via blinking LED indicators. For example, if a drive error (error No. 3) occurs, LED1 and LED2 both start blinking. To clear the error display, perform a reset.

3.3 CPU Block



In this application circuit example, the μ PD70F3116 is used in single-chip mode 0 with an external 5 MHz clock (internal 50 MHz). When using the μ PD703116, replace it with no change. When using the V850E/IA2, set the external 4 MHz (internal 40 MHz) single-chip mode.

3.3.1 Memory map

The memory map is illustrated below.

Figure 3-3. Memory Map (V850E/IA1)

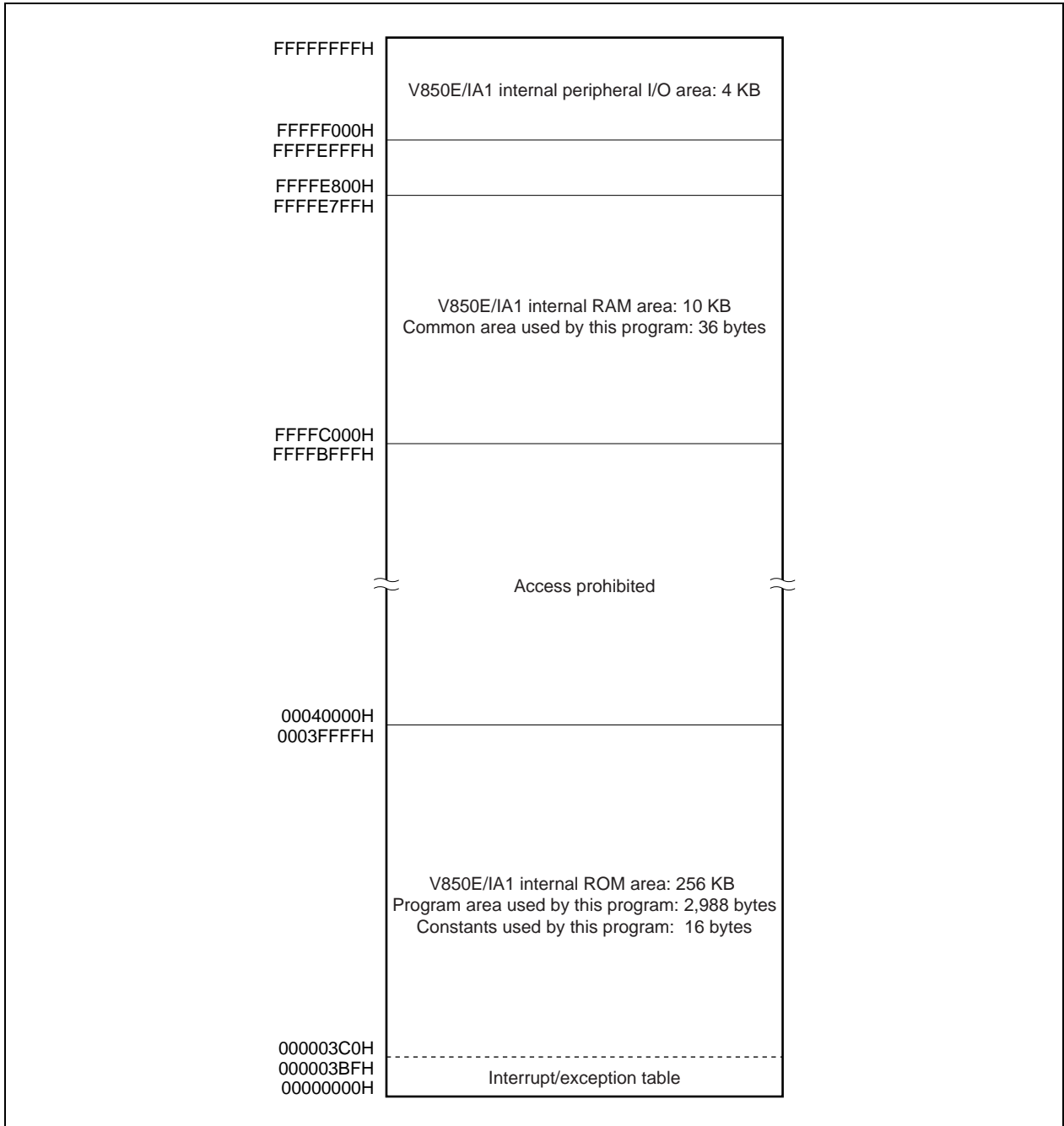
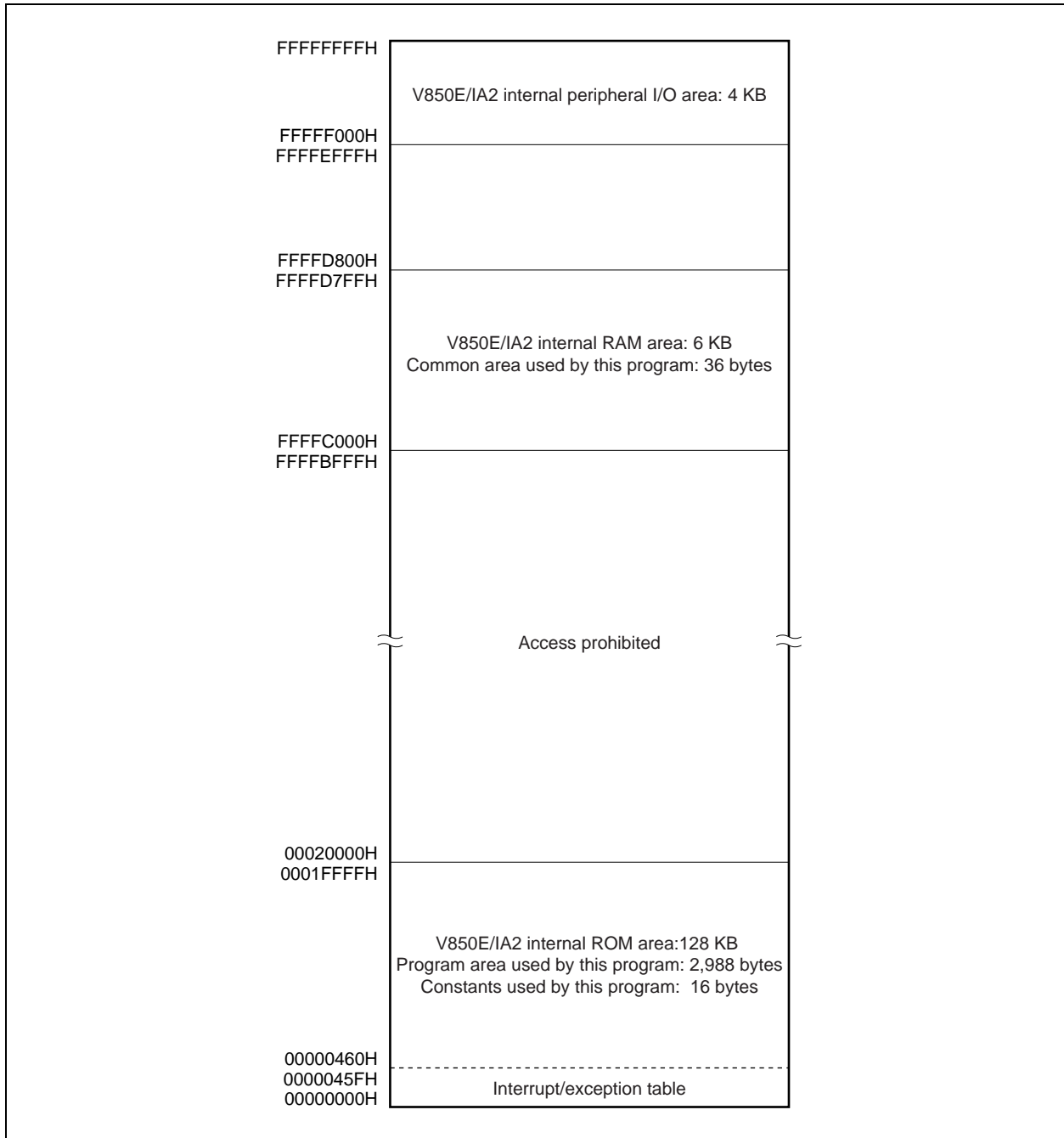


Figure 3-4. Memory Map (V850E/IA2)



3.3.2 Pin assignments

The pin assignment table for the V850E/IA1 is shown below.

Table 3-1. V850E/IA1 Pin Assignment (1/4)

Pin No.	Pin Name	I/O Mode Setting	Signal Name	Active Level
1	ANI07	–	Not used	–
2	AV _{DD}	–	Positive power supply to A/D converter	5 V
3	AV _{SS}	–	Ground potential for A/D converter	0 V
4	AV _{REF1}	Input	Reference voltage input for A/D converter 1	5 V
5	ANI10	Input	V-phase current input	0 to +5 V
6	ANI11	–	Not used	–
7	ANI12	–		–
8	ANI13	–		–
9	ANI14	–		–
10	ANI15	–		–
11	ANI16	–		–
12	ANI17	–		–
13	TRIG_DBG	–		–
14	AD3_DBG	–		–
15	AD2_DBG	–		–
16	AD1_DBG	–		–
17	AD0_DBG	–		–
18	SYNC	–		–
19	CLK_DBG	–		–
20	RESET	Input	Reset input	L
21	CV _{DD}	–	Not used	–
22	CV _{SS}	–		–
23	X1	Input	System clock	–
24	X2	–		–
25	CKSEL	Input	Clock generator operation mode	L
26	MODE0	Input	Operation mode 0	L
27	MODE1	Input	Operation mode 1	H
28	MODE2	Input	Operation mode 2	L
29	P40	Input	Operation mode switch input	L
30	P41	Input		L
31	P42	Output	Watchdog timer output	H/L
32	P43	Input	Drive error input	H
33	P44	Input	Not used	–
34	P45	Input		–
35	P46	Input		–
36	P47	Input		–
37	P30	Output	LED9 output	L
38	P31	Output	LED10 output	L

Remark H: High level

L: Low level

Table 3-1. V850E/IA1 Pin Assignment (2/4)

Pin No.	Pin Name	I/O Mode Setting	Signal Name	Active Level
39	P32	Output	LED11 output	L
40	P33	Output	LED12 output	L
41	P34	Output	LED13 output	L
42	P35	Output	LED14 output	L
43	P36	Output	LED15 output	L
44	P37	Output	LED16 output	L
45	P20	Output	LED1 output	L
46	P21	Output	LED2 output	L
47	P22	Output	LED3 output	L
48	P23	Output	LED4 output	L
49	P24	Output	LED5 output	L
50	P25	Output	LED6 output	L
51	P26	Output	LED7 output	L
52	P27	Output	LED8 output	L
53	V _{DD3}	–	Positive power supply	3.3 V
54	V _{SS3}	–	Ground potential	0 V
55	V _{SS5}	–		0 V
56	V _{DD5}	–		Positive power supply
57	PDL0	Input	Not used	–
58	PDL1	Input		–
59	PDL2	Input		–
60	PDL3	Input		–
61	PDL4	Input		–
62	PDL5	Input		–
63	PDL6	Input		–
64	PDL7	Input		–
65	PDL8	Input		–
66	PDL9	Input		–
67	PDL10	Input		–
68	PDL11	Input		–
69	PDL12	Input		–
70	PDL13	Input		–
71	PDL14	Input		–
72	PDL15	Input		–
73	PDH0	Input		–
74	PDH1	Input		–
75	PDH2	Input		–
76	PDH3	Input		–
77	PDH4	Input		–
78	PDH5	Input		–
79	PDH6	Input		–
80	PDH7	Input		–

Remark H: High level
L: Low level

Table 3-1. V850E/IA1 Pin Assignment (3/4)

Pin No.	Pin Name	I/O Mode Setting	Signal Name	Active Level
81	PCS0	Input	Not used	–
82	PCS1	Input		–
83	PCS2	Input		–
84	PCS3	Input		–
85	PCS4	Input		–
86	PCS5	Input		–
87	PCS6	Input		–
88	PCS7	Input		–
89	V _{PP}	–	Flash write power supply	0 V
90	V _{SS5}	–	Ground potential	0 V
91	V _{DD5}	–	Positive power supply	5 V
92	PCT0	Input	Not used	–
93	PCT1	Input		–
94	PCT2	Input		–
95	PCT3	Input		–
96	PCT4	Input		–
97	PCT5	Input		–
98	PCT6	Input		–
99	PCT7	Input		–
100	PCM0	Input		–
101	PCM1	Input		–
102	PCM2	Input		–
103	PCM3	Input		–
104	PCM4	Input		–
105	TIUD10	Input	Encoder A phase input	L
106	TCUD10	Input	Encoder B phase input	L
107	TCLR10	Input	Encoder Z phase input	L
108	TIUD11	Input	Not used	–
109	TCUD11	Input		–
110	TCLR11	Input		–
111	P00	–		–
112	ESO0	Input	PWM output stopped	H
113	P02	–	Not used	–
114	P03	–		–
115	P04	–		–
116	P05	–		–
117	P06	–		–
118	P07	–		–
119	TO000	Output	U phase output	H
120	TO001	Output	\bar{U} phase output	L
121	TO002	Output	V phase output	H
122	TO003	Output	\bar{V} phase output	L

Remark H: High level
L: Low level

Table 3-1. V850E/IA1 Pin Assignment (4/4)

Pin No.	Pin Name	I/O Mode Setting	Signal Name	Active Level
123	TO004	Output	W phase output	H
124	TO005	Output	\bar{W} phase output	L
125	V _{DD5}	–	Positive power supply	5 V
126	V _{SS5}	–	Ground potential	0 V
127	V _{SS3}	–		0 V
128	V _{DD3}	–	Positive power supply	3.3 V
129	TO010	–	Not used	–
130	TO011	–		–
131	TO012	–		–
132	TO013	–		–
133	TO014	–		–
134	TO015	–		–
135	AV _{DD}	–		Positive power supply to A/D converter
136	AV _{SS}	–	Ground potential for A/D converter	0 V
137	AV _{REF0}	Input	Reference voltage input for A/D converter 0	5 V
138	ANI00	Input	U phase current value input	0 to +5 V
139	ANI01	Input	Speed volume value input	0 to +5 V
140	ANI02	–	Not used	–
141	ANI03	–		–
142	ANI04	–		–
143	ANI05	–		–
144	ANI06	–		–

Remark H: High level

L: Low level

Table 3-2. V850E/IA2 Pin Assignment (1/3)

Pin No.	Pin Name	I/O Mode Setting	Signal Name	Active Level
1	ANI05	–	Not used	–
2	AV _{DD1}	–	Positive power supply to A/D converter	5 V
3	AV _{SS1}	–	Ground potential for A/D converter	0 V
4	ANI10	Input	V-phase current input	0 to +5 V
5	ANI11	–	Not used	–
6	ANI12	–		–
7	ANI13	–		–
8	ANI14	–		–
9	ANI15	–		–
10	ANI16	–		–
11	ANI17	–		–
12	MODE0	Input	Operation mode 0	H
13	V _{SS3}	–	Ground potential	0 V
14	RV _{DD}	–	Positive power supply to regulator	5 V
15	REGOUT	Output	Regulator input	–
16	REGIN	Input	Regulator output	–
17	X1	Input	System clock	–
18	X2	–		–
19	RESET	Input	Reset input	L
20	CV _{SS}	–	Not used	–
21	CKSEL	Input	Clock generator operation mode	L
22	P40	Input	Not used	–
23	P41	Input		–
24	P42	Input		–
25	P30	Input	Operation mode switch input	L
26	P31	Input		L
27	P32	Output	Watchdog timer output	H/L
28	P33	Input	Drive error input	H
29	P34	Input	Not used	–
30	P20	Input		–
31	P21	Input		–
32	P22	Input		–
33	P23	Input		–
34	P24	Input		–
35	P25	Input		–
36	P26	Input		–
37	P27	Input		–
38	V _{SS}	–	Ground potential	0 V
39	V _{DD}	–	Positive power supply	5 V

Remark H: High level

L: Low level

Table 3-2. V850E/IA2 Pin Assignment (2/3)

Pin No.	Pin Name	I/O Mode Setting	Signal Name	Active Level
40	PDL0	Output	LED1 output	L
41	PDL1	Output	LED2 output	L
42	PDL2	Output	LED3 output	L
43	PDL3	Output	LED4 output	L
44	PDL4	Output	LED5 output	L
45	PDL5	Output	LED6 output	L
46	PDL6	Output	LED7 output	L
47	PDL7	Output	LED8 output	L
48	PDL8	Output	LED9 output	L
49	PDL9	Output	LED10 output	L
50	PDL10	Output	LED11 output	L
51	PDL11	Output	LED12 output	L
52	PDL12	Output	LED13 output	L
53	PDL13	Output	LED14 output	L
54	PDL14	Output	LED15 output	L
55	PDL15	Output	LED16 output	L
56	PDH0	Input	Not used	–
57	PDH1	Input		–
58	PDH2	Input		–
59	PDH3	Input		–
60	PDH4	Input		–
61	PDH5	Input		–
62	V _{PP}	–	Flash write power supply	0 V
63	V _{SS3}	–	Ground potential	0 V
64	V _{DD}	–	Positive power supply	5 V
65	PCT0	Input	Not used	–
66	PCT1	Input		–
67	PCT4	Input		–
68	PCT6	Input		–
69	PCM0	Input		–
70	PCM1	Input		–
71	TIUD10	Input	Encoder A phase input	L
72	TCUD10	Input	Encoder B phase input	L
73	TCLR10	Input	Encoder Z phase input	L
74	P00	–	Not used	–
75	ESO0	Input	PWM output stopped	H
76	P02	–	Not used	–
77	P03	–		–
78	P04	–		–
79	P05	–		–

Remark H: High level
L: Low level

Table 3-2. V850E/IA2 Pin Assignment (3/3)

Pin No.	Pin Name	I/O Mode Setting	Signal Name	Active Level
80	TO000	Output	U phase output	H
81	TO001	Output	\overline{U} phase output	L
82	TO002	Output	V phase output	H
83	TO003	Output	\overline{V} phase output	L
84	TO004	Output	W phase output	H
85	TO005	Output	\overline{W} phase output	L
86	V _{DD}	–	Positive power supply	5 V
87	V _{SS}	–	Ground potential	0 V
88	TO010	–	Not used	–
89	TO011	–		–
90	TO012	–		–
91	TO013	–		–
92	TO014	–		–
93	TO014	–		–
94	AV _{DD0}	–		Positive power supply to A/D converter
95	AV _{SS0}	–	Ground potential for A/D converter	0 V
96	ANI00	Input	U phase current value input	0 to +5 V
97	ANI01	Input	Speed volume value input	0 to +5 V
98	ANI02	–	Not used	–
99	ANI03	–		–
100	ANI04	–		–

Remark H: High level
L: Low level

3.3.3 Peripheral I/O

The following types of peripheral I/O functions are used in this application circuit.

★ **Table 3-3. List of Peripheral I/O Functions**

Peripheral I/O Function (V850E/IA1)	Peripheral I/O Function (V850E/IA2)	Description
P20 to P27	PDL0 to PDL15	LED output
P30 to P37		
P40, P41	P30, P31	Operation mode switch input
P42	P32	Watchdog timer output
P43	P33	Drive error input
Timer 00 (TM00)		PWM output
Timer 10 (TM10)		Encoder counter
Timer 3 (TM3)		10 ms interval timer
Timer 4 (TM4)		0.4 ms interval timer
ANI00		U phase current value input
ANI01		Speed volume value input
ANI10		V phase current value input

(1) Description of peripheral I/O functions

(a) LED output

★ LED output uses P20 to P27 and P30 to P37 for the V850E/IA1, and PDL0 to PDL15 for the V850E/IA2.

- LED display method

LEDs light to indicate when a bit value is "0" (zero).

For the V850E/IA1, pins P20 to P27 (LED1 to LED8) and pins P30 to P36 (LED9 to LED15) indicate absolute numerical values and P37 (LED16) indicates the minus sign.

★ For the V850E/IA2, pins PDL0 to PDL14 (LED1 to LED15) indicate absolute numerical values and PDL15 (LED16) indicates the minus sign.

0: Lit (ON)

1: Not lit (OFF)

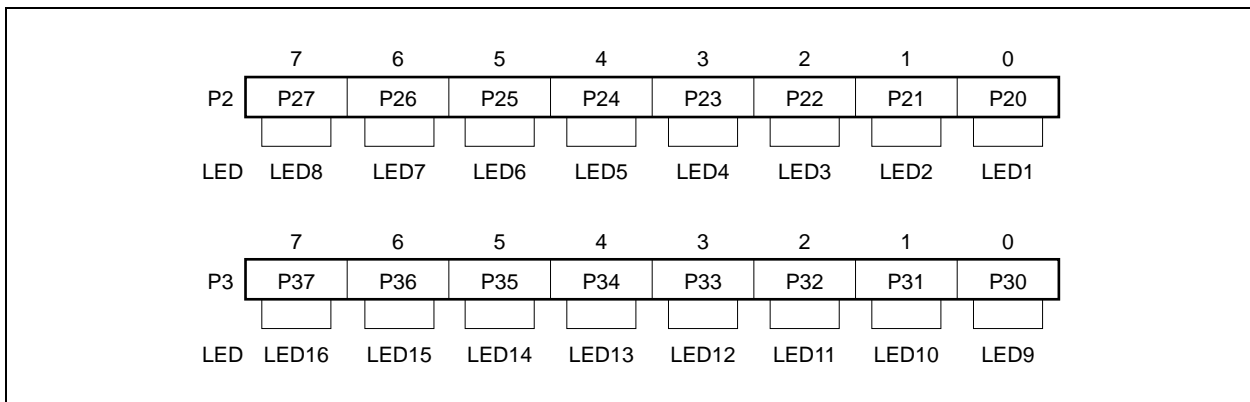
- LED displays

Rotation when operation mode is STOP operation mode: Indicates differential between target position and current position

Rotation when operation mode is clockwise or counter-clockwise operation mode:

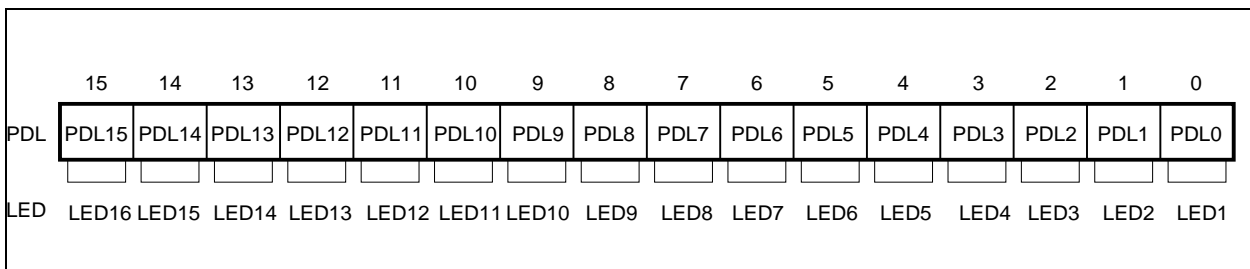
Indicates pulse count changed by 0.4 ms interrupt (speed differential)

Figure 3-5. LED Display (V850E/IA1)



★

Figure 3-6. LED Display (V850E/IA2)



★

(b) Operating mode switch input, watchdog timer output, drive error input

These I/O use port 4 in the V850E/IA1 and port 3 in the V850E/IA2.

The function of each of the bits of port 4 or port 3 is shown below.

Figure 3-7. Functions of Port 4 (V850E/IA1)

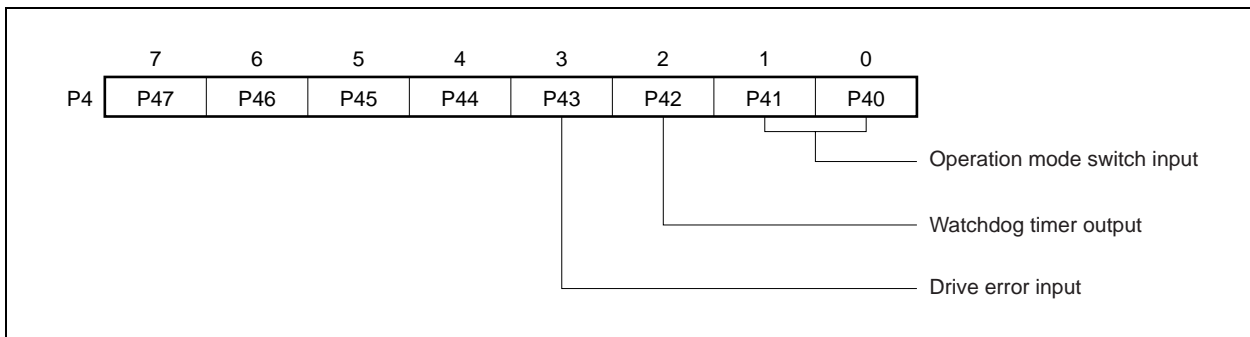
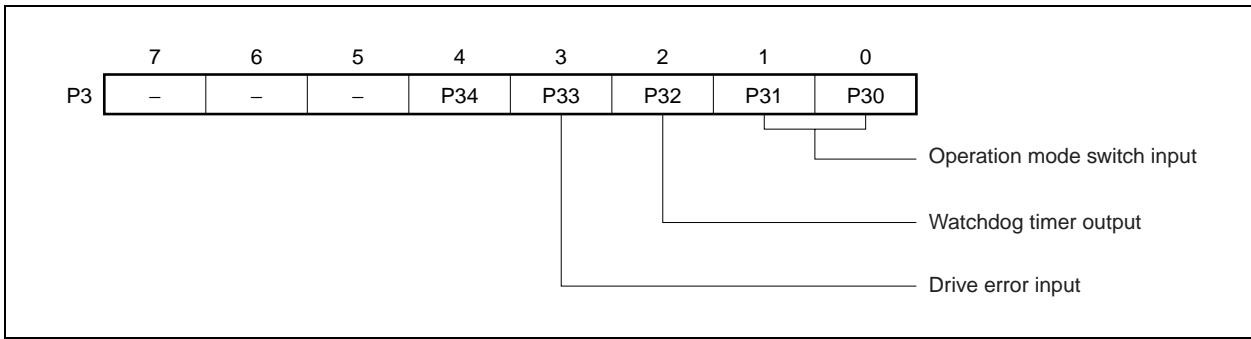




Figure 3-8. Functions of Port 3 (V850E/IA2)



- P41 and P40 (V850E/IA1), or P31, P30 (V850E/IA2): Operation mode switch input
Bit 1 (Pn1) and bit 0 (Pn0) specify the operation mode.

Pn1	Pn0	Operation Mode
0	0	STOP operation
1	1	
0	1	Clockwise rotation operation
1	0	Counter-clockwise rotation operation

- P42 (V850E/IA1) or P32 (V850E/IA2): Watchdog timer output
By setting bit 2 (Pn2), pulses that are generated every 0.4 ms using timer 4 (TM4) are output by software to the watchdog timer circuit. If a pulse is not output after 1 ms or more, the watchdog timer circuit sends a PWM stop instruction to the ESO0 pin of the V850E/IA1 or V850E/IA2.

- P43 (V850E/IA1) or P33 (V850E/IA2): Drive error input
Bit 3 (Pn3) indicates the driver error input status.

Pn3	Operation
0	Normal
1	Drive error

When an error signal occurs in the driver (i.e., when the Pn3 bit = 1), PWM is stopped and notification is sent to the CPU.

Remark V850E/IA1: n = 4
V850E/IA2: n = 3

(c) PWM output

Timer 00 (TM00) is used to output PWM waveforms. In this application circuit example, the settings are as shown below.

- 20 kHz symmetrical triangular waveform mode
- Dead time: 20 μ s
- TO000 to TO005: Log active
- When ESO0 pin input is at high level, PWM output is stopped.

(d) Encoder input

Encoder input is counted using timer 10 (TM10). In this application circuit example, the settings are as shown below.

- Up/down counter mode's UDC mode A
- Counter is cleared after falling edge of TCLR10 is detected
- $\times 4$ frequency multiplication (mode 4) is used

(e) 10 ms timer interrupt

Timer 3 (TM3) is used to issue interrupts at a 10 ms interval.

(f) 0.4 ms timer interrupt

Timer 4 (TM4) is used to issue interrupts at a 0.4 ms interval.

(g) Current value input

ANI00: U phase current value (–5 to +5 A)
ANI10: V phase current value (–5 to +5 A)

(h) Speed specification volume value input

ANI01 is used to input a value from 0 to 1,023.

3.4 Circuit Diagram

Figure 3-9 shows a diagram of the application circuit example.

This application circuit diagram includes the V850E/IA1 (μ PD70F3116GJ-UEN) a reset circuit, oscillator, a pin handling microcontroller peripheral block, operation mode switch block, LED output block, watchdog timer circuit block, drive circuit block, motor controller, and motor rotation indicator.

(1) Microcontroller and microcontroller peripheral block

The V850E/IA1 includes a reset circuit, an oscillation circuit that uses a 5 MHz resonator, and handling of a MODE pin and unused pins.

★

A 4 MHz oscillator is used for the V850E/IA2.

(2) Operation mode switch block

This includes switches that set the operation mode as clockwise or counter-clockwise operation.

(3) LED output block

This block includes 16 LEDs, which are used to indicate rotation speeds (rpm), errors, etc.

(4) Watchdog timer circuit block

This block uses the μ PD74HC123A to output stop signals when pulse output from the V850E/IA1 stops for at least one ms.

(5) Drive circuit block

The 6-phase outputs from TO000 to TO006 are converted to U-, V-, and W-phase output for the motor driver. This drive circuit is not shown in detail in this example, since it varies depending on the motor's specifications.

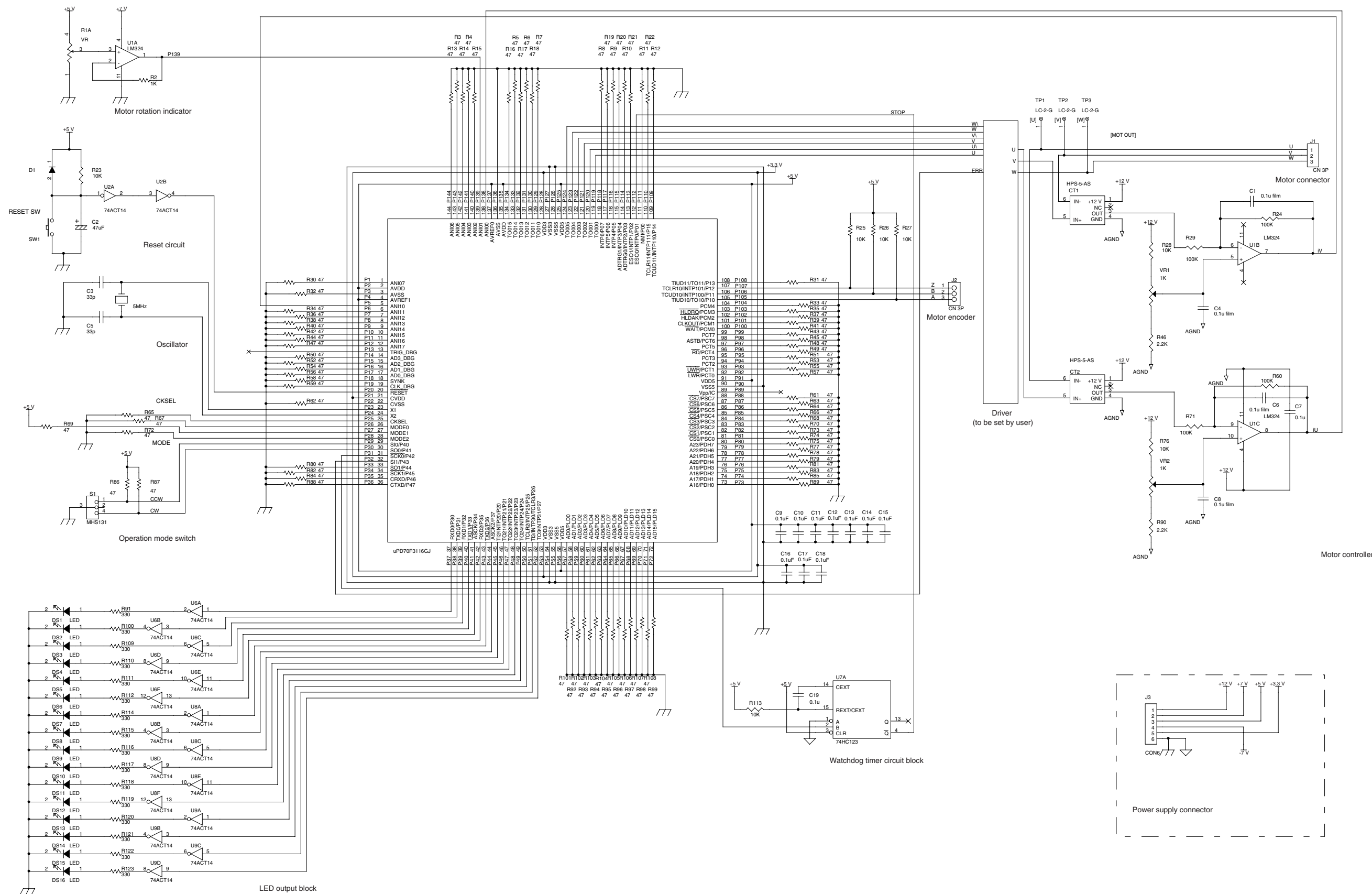
(6) Motor controller

This block includes the HPS-5-AS, LM324, and other devices that are used to measure the motor's drive currents U and V via A/D conversion.

(7) Motor rotation indicator

This block includes a volume adjuster and the LM324 for setting the motor's rotation speed (rpm).

Figure 3-9. Circuit Diagram of Application Circuit Example



CHAPTER 4 CONTROL SYSTEM

4.1 Overview

4.1.1 Control principles

Typically, when controlling a 3-phase motor, voltage and current are indicated as 3-phase AC. However, 2-phase AC is easier to represent than 3-phase AC. Also, control is even simpler when representing biaxial DC rather than 2-phase AC.

When converting biaxial DC (d-q axes), numerous armature coils are connected to the commutator and wound in the radial direction, similar to a DC motor such as is shown in part (c) of Figure 4-1 below. Two voltage values, v_d (d-axis voltage) and v_q (q-axis voltage), are applied and two currents, i_d (d-axis current) and i_q (q-axis current), flow across the brush that is situated along the d-q axis which rotates at the same speed as the electromagnetic field.

In the application circuit example, the current and voltage are controlled along the d-q axis in the same way as in DC motors.

Figure 4-1. Equivalent Circuits (1/3)

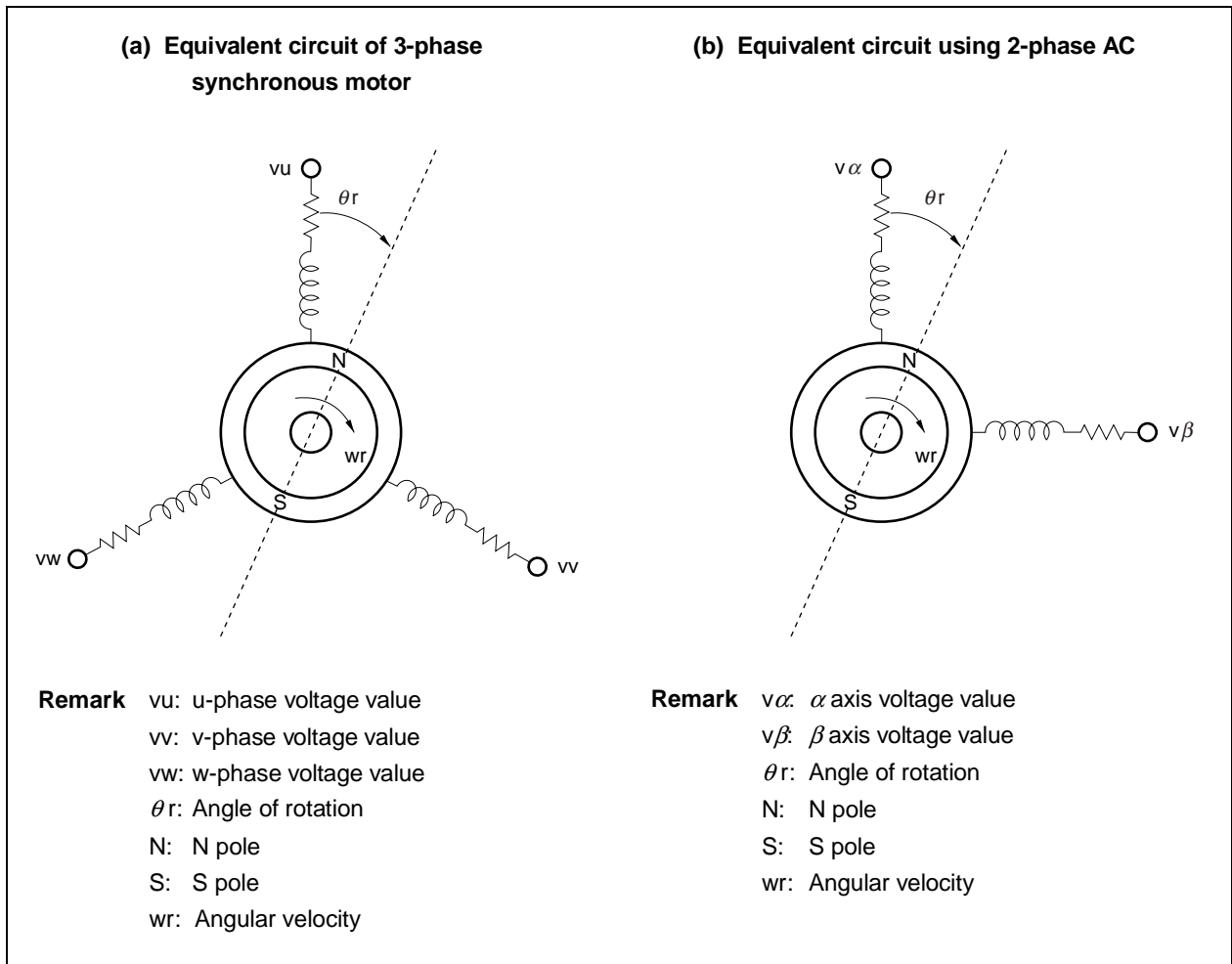
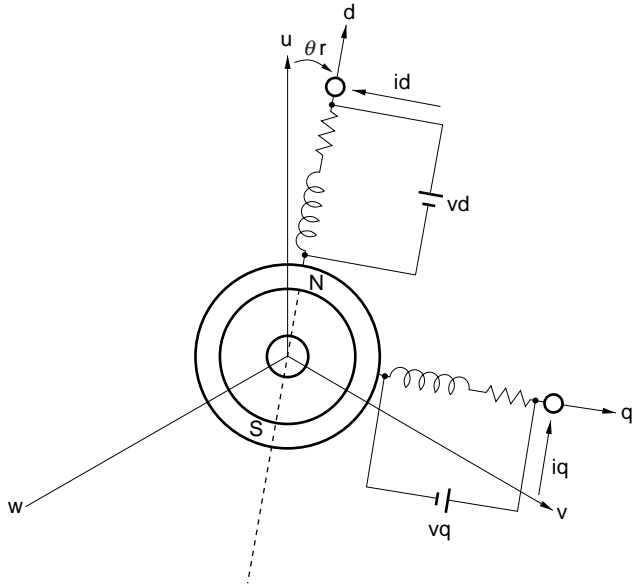


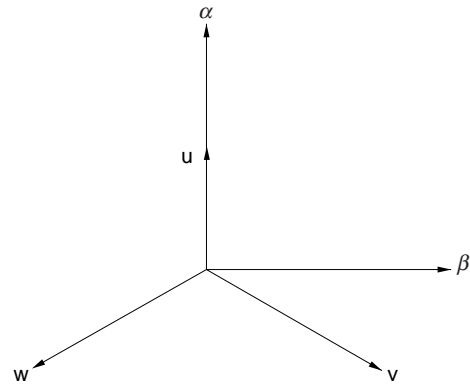
Figure 4-1. Equivalent Circuits (2/3)

(c) Equivalent circuit of biaxial DC type, (d-q) axes



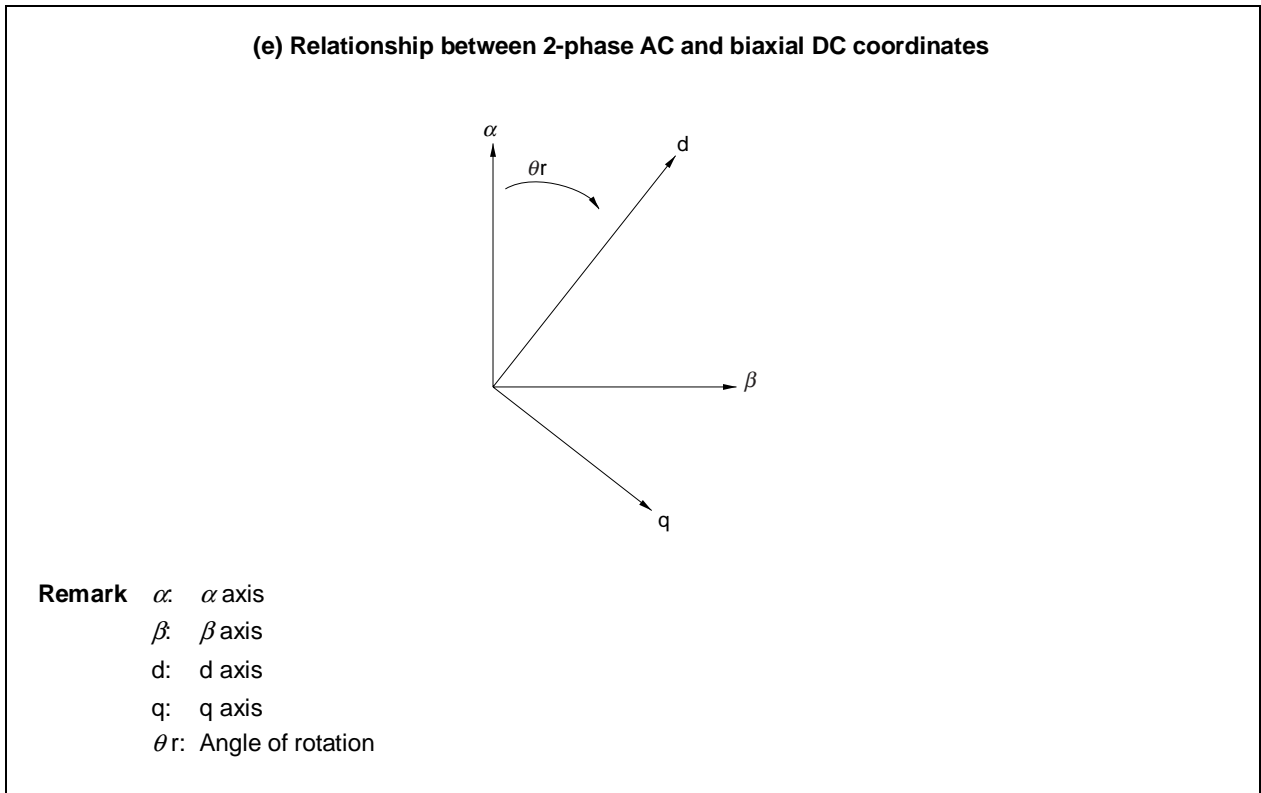
Remark u: u phase
 v: v phase
 w: w phase
 θ_r : Angle of rotation
 N: N pole
 S: S pole
 d: d axis
 id: d-axis current value
 vd: d-axis voltage value
 q: q axis
 iq: q-axis current value
 vq: q-axis voltage value

(d) Relationship between 3-phase and 2-phase AC coordinates



Remark u: u phase
 v: v phase
 w: w phase
 α : α axis
 β : β axis

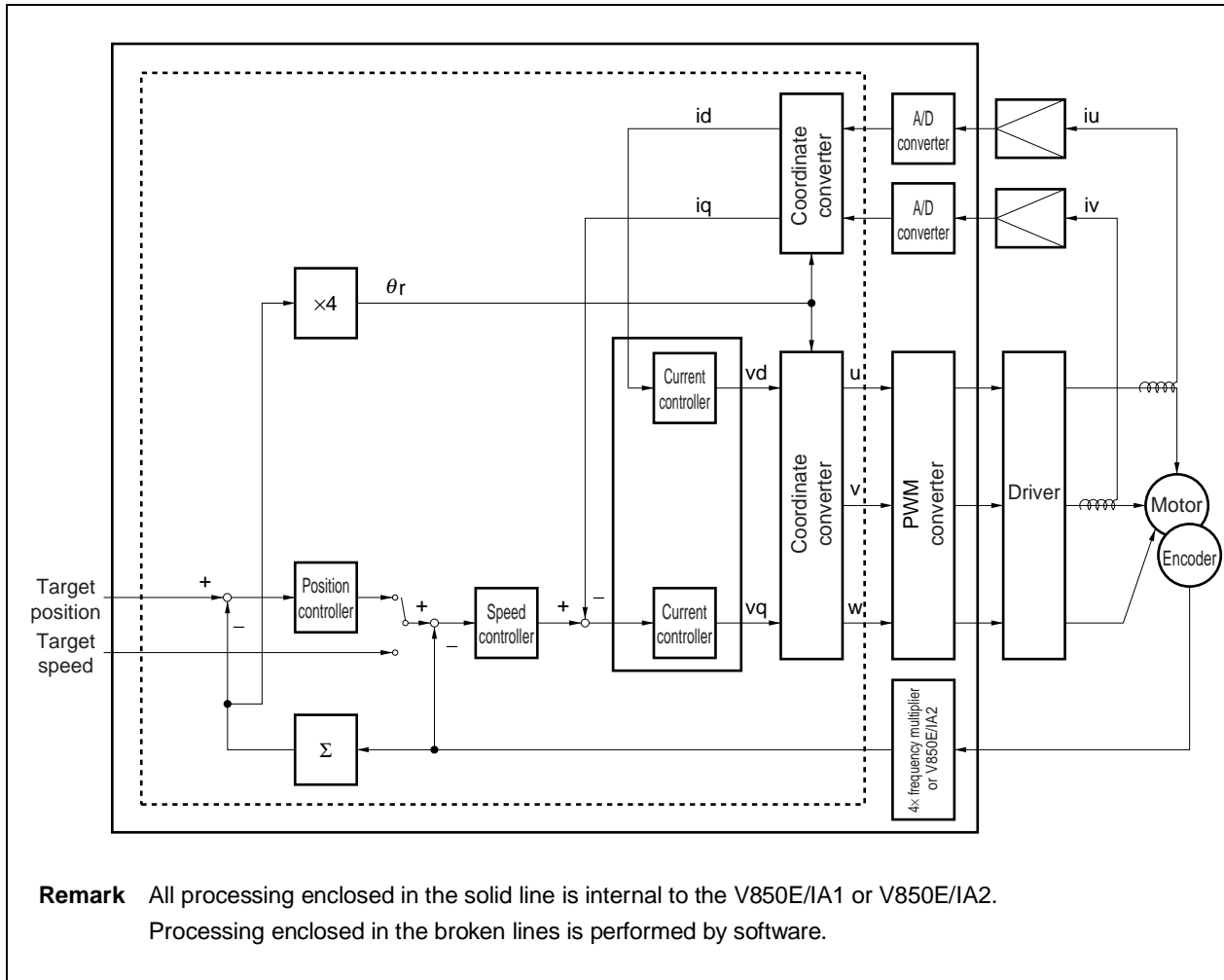
Figure 4-1. Equivalent Circuits (3/3)



4.1.2 Control block

★ In the application circuit example, processing of timer interrupts in 0.4 ms units is used to calculate control along the d-q axis (as is shown in Figure 4-2) and final output of the u, v, and w phase voltage values is performed by the PWM timer function (timer 00 [TM00]) of V850E/IA1 or V850E/IA2.

Figure 4-2. Control Block



★ **Remark** All processing enclosed in the solid line is internal to the V850E/IA1 or V850E/IA2. Processing enclosed in the broken lines is performed by software.

In the application circuit example, the motor is controlled by entering a target position or target speed.

(1) Position control

If a target position has been set, speed conversion is performed based on the differential between the target position and the current position.

(2) Speed control

The target current value is calculated based on the differential between the target speed (previously set or calculated) and the current speed.

(3) Current control

The target voltage for the d-q axis is calculated based on the current return value converted for the d-q axis and the target current value.

(4) Coordinate conversion

The voltage along the d-q axis is converted to 3-phase AC voltage.

(5) PWM conversion

The on-chip PWM function of the V850E/IA1 or V850E/IA2 is used to perform PWM output of the calculated 3-phase AC voltage.

★

4.1.3 Motor specifications

The specifications for the motor used in the application circuit example are listed in Table 4-1.

Table 4-1. Motor Specifications

Item	Three-Phase Synchronous Motor
Rated output	50 W
Drive power supply voltage	100 V
Rated torque	1.62 kg·f·cm
Instantaneous maximum torque	4.9 kg·f·cm
Rated count	3,000 rpm
Maximum count	5,000 rpm
Encoder	Incremental 2,500/r (A phase, B phase, Z phase)
Pole count	4 poles
Offset from Z phase	200 pulses

4.2 Position Control

The following equation expresses the conversion from position to speed.

$$o_speed = kp \times (o_position - now_position)$$

Remark	o_speed:	Target speed
	kp:	Position-proportional gain
	o_position:	Target position
	now_position:	Current position

4.3 Speed Control

In the application circuit example, PI (Proportion, Integral) control is used in the speed control block. The equations used for speed control are shown below.

$$d_speed = o_speed - now_speed$$

$$o_iqp = ksp \times d_speed$$

$$o_iqi(n) = o_iqi(n-1) + (ksi \times d_speed(n-1))$$

$$o_iq = o_iqp + o_iqi(n)$$

Remark	d_speed:	Differential between target speed and current speed
	o_speed:	Target speed
	now_speed:	Current speed
	o_iqp:	Speed-proportional component current value
	ksp:	Speed-proportional gain
	o_iqi:	Speed-integral component current value
	ksi:	Speed-integral gain
	o_iq:	Target current value
	n:	Current component
	n-1:	Previous component

4.4 Current Control

For current control, the d-axis current (i_d) and q-axis current (i_q) are converted via the following equations to obtain a target voltage for each axis.

$$o_vd = k_i \times (-i_d)$$

$$o_vq = k_i \times (o_iq - i_q)$$

Remark o_vd : Target d-axis voltage
 k_i : Current-proportional gain
 i_d : d-axis current value
 o_vq : Target q-axis voltage
 o_iq : Target q-axis current value
 i_q : q-axis current value

i_d and i_q are obtained by converting current values for the u and v phases to d-q axis coordinates. The equations are shown below.

$$i_d = i_v \times \cos \theta_r - i_u \times \cos (\theta_r - 2\pi/3)$$

$$i_q = i_v \times \sin \theta_r - i_u \times \sin (\theta_r - 2\pi/3)$$

Remark i_d : d-axis current value
 i_q : q-axis current value
 i_u : u-phase current value
 i_v : v-phase current value
 θ_r : Angle of rotation

4.5 Three-Phase Voltage Conversion

The equations used to convert voltage values (v_d and v_q) calculated for the d-q axis to 3-phase coordinates are shown below.

$$o_vu = o_vd \times \cos \theta_r - o_vq \times \sin \theta_r$$

$$o_vv = o_vd \times \cos (\theta_r - 2\pi/3) - o_vq \times \sin (\theta_r - 2\pi/3)$$

$$o_vw = -o_vu - o_vv$$

Remark o_vu : Target u-phase voltage
 o_vv : Target v-phase voltage
 o_vw : Target w-phase voltage
 o_vd : Target d-axis voltage
 o_vq : Target q-axis voltage
 θ_r : Angle of rotation

4.6 PWM Conversion

- ★ The calculated target voltage is output by a 16-bit timer (TM00) that is used for the 3-phase sine-wave PWM inverter of the V850E/IA1 or V850E/IA2 (see **5.1 PWM Timer Function (Timer 00 [TM00])**).

4.7 Encoder Input Processing

- ★ For $\times 4$ frequency multiplication, the encoder uses a 16-bit up/down counter (TM10) that can be used for 2-phase encode input/general timer functions in the V850E/IA1 or V850E/IA2. See **5.2 Encoder Counter Functions (Timer 10 [TM10])**.

In the control system used in the application circuit example, absolute positions along the motor axes must be detected, so encoder values are cleared in the Z phase and absolute positions are detected. Processing beyond the Z phase is performed by software, with 32-bit position control (see **7.2 Motor Control Interrupt Servicing (0.4 ms Interval)**).

CHAPTER 5 FUNCTIONS OF V850E/IA1 AND V850E/IA2

This chapter describes the use of the timer that is used for PWM output and encoder input.

5.1 PWM Timer Function (Timer 00 [TM00])



The V850E/IA1 and V850E/IA2 include a 2-channel, 3-phase PWM output function that uses an optimum dead time for motor control applications. This function can be used to control two motors at the same time. It features three waveform modes as comparative waveforms for PWM generation.

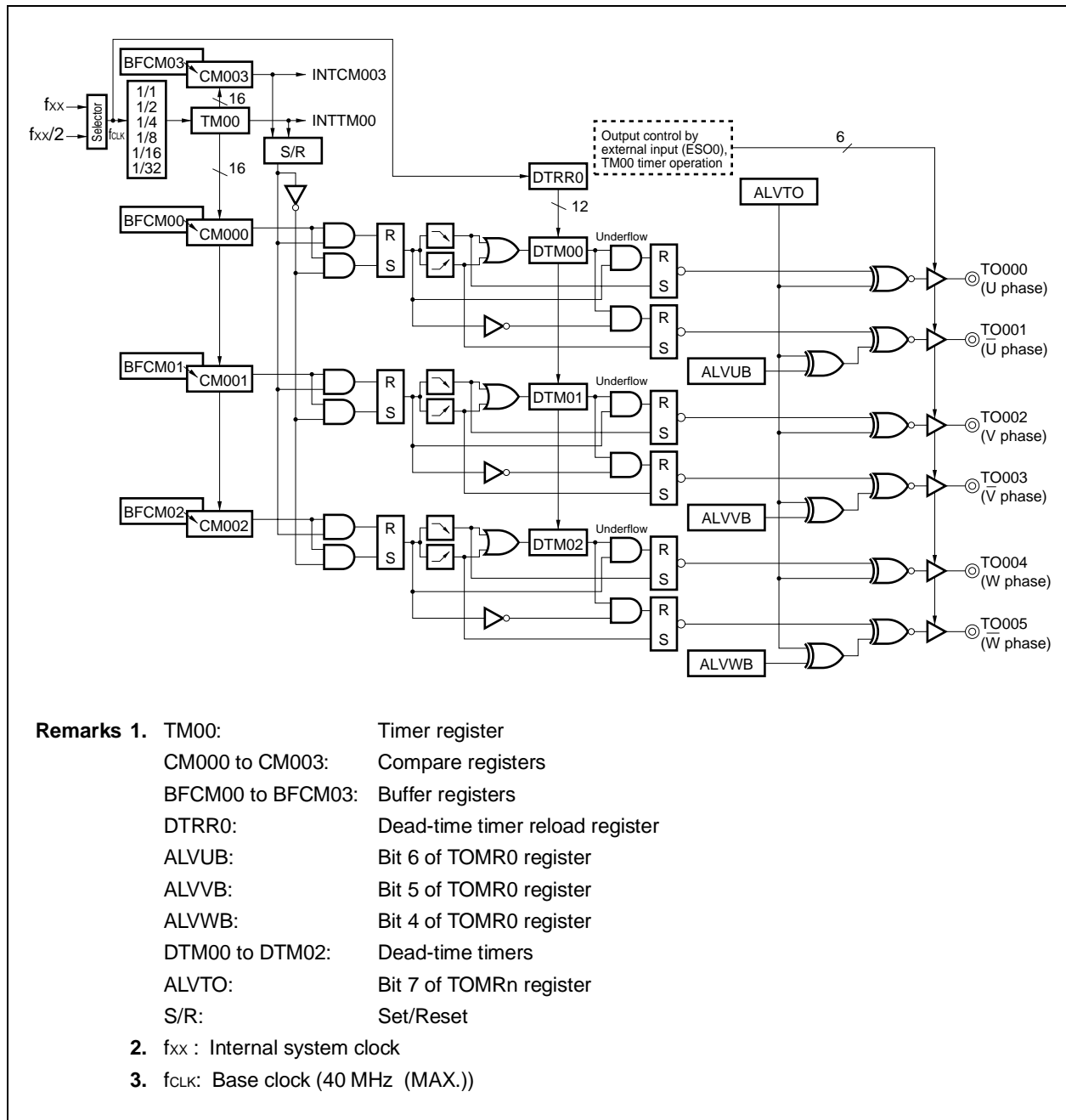
- PWM mode 0: Symmetrical triangular waves
- PWM mode 1: Asymmetrical triangular waves
- PWM mode 2: Saw-tooth waves

In this application circuit example, timer 00 (TM00) is used on one channel in PWM mode 0 (symmetrical triangular waves). Only one motor is used in this case.

5.1.1 General

This section shows a block diagram of each mode and provides general descriptions of registers.

Figure 5-1. Block Diagram of Timer 00 (TM00) (PWM Mode 0: Symmetric Triangular Wave)



(1) Timer 00 (TM00)

The TM00 register operates as a 16-bit up/down timer or up timer. The cycle is controlled by compare register 003 (CM003).

TM00 register start/stop is controlled by the TM0CE0 bit of timer control register 00 (TMC00).

Division by the prescaler is set for the count clock of this application circuit example to f_{CLK} with the PRM02 to PRM00 bits of the TMC00 register (f_{CLK} : base clock, see **5.1.3 (1) Timer 00 clock selection register (PRM01) settings**).

The conditions when the TM00 register becomes 0000H are as follows.

- Reset input
- TM0CE0 bit = 0
- Immediately after overflow or underflow

(2) Dead-time timers 00 to 02 (DTM00 to DTM02)

The DTM00 to DTM02 registers are dedicated 12-bit down timers that generate dead time suitable for inverter control applications. DTM00 to DTM02 operate as one-shot timer.

Counting by a dead-time timer is enabled or disabled by the TM0CED0 bit of timer control register 00 (TMC00) and cannot be controlled through software. Dead-time timer count start and stop is controlled through hardware.

A dead-time timer starts counting down when the value of dead-time timer reload register 0 (DTRR0) is transferred in synchronization with the compare match timing of compare registers 000 to 002 (CM000 to CM002).

When the value of a dead-time timer changes from 000H to FFFH, the dead-time timer generates an underflow signal, and the timer stops at the value FFFH.

If the value of a dead-time timer matches the value of the corresponding compare register before underflow of the dead-time timer takes place, the value of the DTRR0 register is transferred to the dead-time timer again, and the timer starts down counting.

The count clock of the dead-time timer is fixed to the base clock (f_{CLK}), and the dead-time width is (set value of DTRR0 register + 1)/base clock (f_{CLK}).

If TM00 operates in PWM mode 0 with the dead-time timer count operation disabled, an opposite signal without dead time is output to TO000 and TO001, TO002 and TO003, and TO004 and TO005.

(3) Dead-time timer reload register 0 (DTRR0)

DTRR0 register is a 12-bit register used to set the values of the three dead-time timers (DTM00 to DTM02 registers). However, a value is transferred from the DTRR0 register to each dead time register independently.

DTRR0 can be read/written in 16-bit units. All 0s are read for the higher 4 bits when 16-bit read access is performed to the DTRR0 register.

- Cautions**
1. **Changing the value of the DTRR0 register during TM00 operation (TM0CE0 bit of TMC00 register = 1) is prohibited.**
 2. **Be sure to write 0 in the higher 4 bits.**

(4) Compare registers 000 to 002 (CM000 to CM002)

The CM000 to CM002 registers are 16-bit registers that always compare their own values with the value of the TM00 register. If the value of a compare register matches the value of TM00, the compare register outputs a trigger signal, and changes the contents of the flip-flop (F/F) connected to the compare register. Each of the CM000 to CM002 registers is provided with a buffer register (BFCM00 to BFCM02), so that the contents of the

buffer are transferred to the CM000 to CM002 registers at the following base clock (f_{CLK}). Transfer is enabled or disabled by the BFTEN bit of the TMC00 register.

(5) Compare register 003 (CM003)

The CM003 register is a 16-bit register that always compares its value with the value of TM00. If the values match, CM003 outputs an interrupt signal (INTCM003). The CM003 register controls the maximum count value of the TM00 register, and if the values match, it performs the following operations at the next timer count clock.

- In PWM mode 0: Switches TM00 operation from up count to down count

The CM003 register also has a buffer register (BFCM03) and transfers the buffer contents in the next base clock (f_{CLK}) cycle to the CM003 register. Transfer enable or disable is controlled with the BFTE3 bit of the TMC00 register.

(6) Buffer registers CM00 to CM02 (BFCM00 to BFCM02)

The BFCM00 to BFCM02 registers are 16-bit registers that transfer data to the compare register (CM000 to CM002) corresponding to each buffer register when an interrupt request signal (INTCM003/INTTM00) is generated.

BFCM00 to BFCM02 can be read/written in 16-bit units.

Caution The set values of the BFCM00 to BFCM02 registers are transferred to the CM000 to CM002 registers at the following timing.

- When TM0CE0 bit of TMC00 register = 0: Transfer at next operation timing after write to BFCM00 to BFCM02 register
- When TM0CE0 bit of TMC00 register = 1: Value of BFCM00 to BFCM02 registers is transferred to CM000 to CM002 registers upon occurrence of INTTM00 or INTCM003. At this time, transfer enable or disable is controlled by the BFTEN bit of the timer control register (TMC00).

(7) Buffer register CM03 (BFCM03)

The BFCM03 register is a 16-bit register that transfers data to the compare register at any timing. Transfer enable or disable is controlled by the BFTE3 bit of the TMC00 register.

BFCM03 can be read/written in 16-bit units.

Cautions 1. The set value of the BFCM03 register is transferred to the CM003 register at the following timing.

- When TM0CE0 bit of TMC00 register = 0: Transfer at next operation timing after write to BFCM03 register
- When TM0CE0 bit of TMC00 register = 1: Value of BFCM03 register is transferred to CM003 register upon occurrence of INTTM00. At this time, transfer enable or disable is controlled by the BFTE3 bit of the timer control register (TMC00).

2. Setting the BFCM03 register to 0000H is prohibited.

5.1.2 Use of PWM timer in application circuit example

(1) Determination of PWM frequency

The following factors are taken into consideration when determining the PWM frequency for motor control.

- Drive circuit's switching time
- Dead time (dead frequency band)
- Choke coil noise

- Cautions**
1. If the drive circuit's switching response is poor, it cannot be used for high frequencies.
 2. The portion occupied by dead time may be a problem when attempting to achieve high-precision motor control.
 3. When a choke coil is used, noise is heard if the PWM cycle is within the audible frequency range.
 4. If a very high frequency is selected, it becomes difficult to achieve good resolution based on the timer setting.

The PWM timer's cycle is determined based on the timer's input clock (basic clock) and the frequency division ratio (count clock). The PWM timer's cycle setting is performed using the timer 00 clock select register (PRM01) and timer control register 00 (TMC00).

Once the count clock has been determined, use the BFCM03 register to set the CM003 comparison value. This comparison value is one-half of the cycle for PWM mode 0 (symmetrical triangular wave mode).

In this application circuit example, the PWM cycle is 20 kHz and the dead time is 2 μ s. Therefore, the following values are determined when the system clock frequency is 50 MHz.

Input clock (basic clock) selection:	$f_{xx}/2$ (PRM1 bit of PRM01 register = 0)
Division ratio:	1/1 (PRM02 to PRM00 bits of TMC00 register = 000)
Count clock:	25 MHz
CM003 value:	$(\text{Count clock frequency}/\text{PWM frequency}) \times 1/2 = 625$

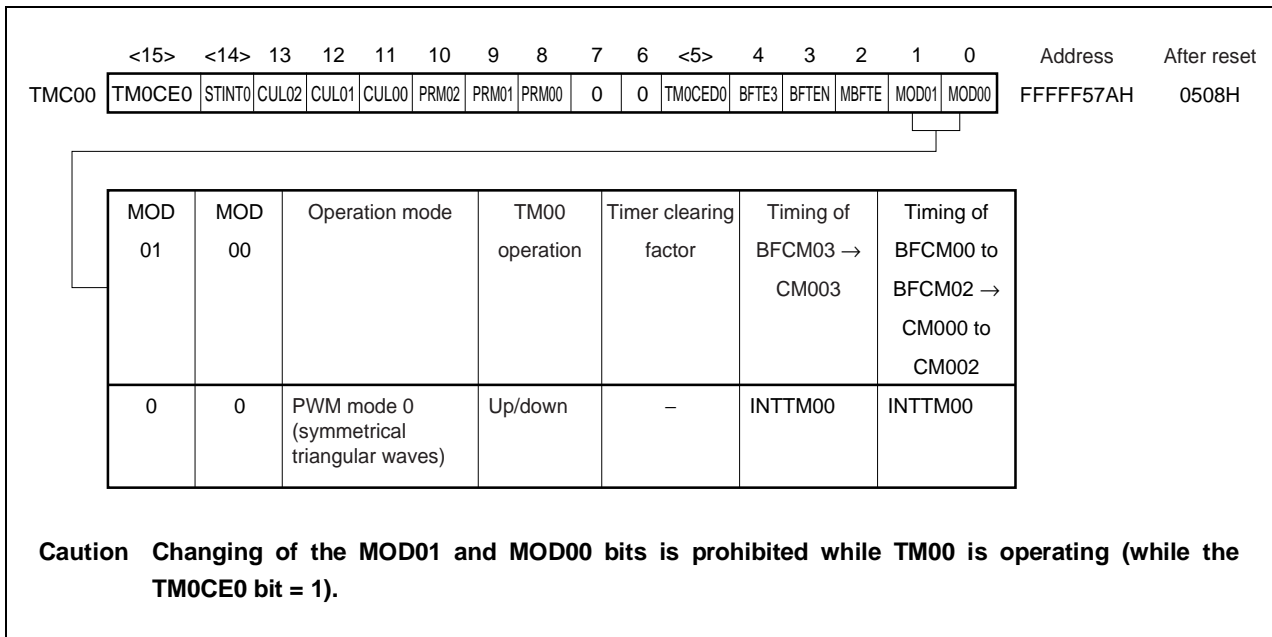
In the case of the V850E/IA2, to set the PWM frequency to 20 kHz and the dead time to 2 μ s, the following values are used with the system clock set to 40 MHz.

Input clock (basic clock) selection:	$f_{xx}/2$ (PRM1 bit of PRM01 register = 1)
Division ratio:	1/1 (PRM02 to PRM00 bits of TMC00 register = 000)
Count clock:	40 MHz
CM003 value:	$(\text{Count clock frequency}/\text{PWM frequency}) \times 1/2 = 1000$

(2) Determination of operation mode

The TMC00 register is used to set the operation mode for the PWM timer (TM00), as is shown in Figure 5-2.

Figure 5-2. Operation Mode Setting



Although PWM mode 0 (symmetrical triangular wave mode) is used in this application circuit example, the optimum operation mode for the target system can be selected.

The next section describes the output waveform widths that correspond to the PWM mode 0 operation and settings.

[Operation]

In PWM mode 0, TM00 performs up/down count operations. When TM00 = 0000H during down counting, an underflow interrupt (INTTM00) is generated, and when TM00 = CM003 during up counting, a match interrupt (INTCM003) is generated.

Switching from up counting to down counting is performed when TM00 and CM003 match (INTCM003), and switching from down counting to up counting is performed when TM00 underflow occurs after TM00 becomes 0000H.

The PWM cycle in this mode is (BFCM03 value \times 2 \times TM00 count clock). Concerning setting of data to BFCM03, the next PWM cycle width is set to BFCM03.

The data of BFCM03 is automatically transferred by hardware to CM003 upon generation of the INTTM00 interrupt. Furthermore, calculation is performed by software processing started by INTTM00, and the data for the next cycle is set to BFCM03.

Data setting to CM000 to CM002, which control the PWM duty, is explained next.

Setting of data to CM000 to CM002 consists in setting the duty output from BFCM00 to BFCM02.

The values of BFCM00 to BFCM02 are automatically transferred by hardware to CM000 to CM002 upon generation of the INTTM00 interrupt. Furthermore, software processing is started up and calculation performed, and set/reset timing of the F/F for the next cycle is set to BFCM00 to BFCM02.

The PWM cycle and the PWM duty are set in the above procedure.

The F/F set/reset conditions upon match of CM000 to CM002 are as follows.

- Set: CM000 to CM002 match detection during TM00 up count operation
- Reset: CM000 to CM002 match detection during TM00 down count operation

In this mode, the F/F set/reset timing is performed in the same timing (right-left symmetric control). The values of DTRR0 are transferred to the corresponding dead-time timers (DTM00 to DTM02) in synchronization with the set/reset timing of the F/F, and down counting is started. DTM00 to DTM02 count down to 000H, and stop when they count down further to FFFH.

DTM00 to DTM02 can automatically generate a width (dead time) at which the active levels of the positive phase (TO000, TO002, TO004) and negative phase (TO001, TO003, TO005) do not overlap.

In this way, software processing is started by an interrupt (INTTM00) that occurs once during every PWM cycle after initial setting has been performed, and by setting the PWM cycle and PWM duty to be used in the next cycle, it is possible to automatically output a PWM waveform to TO000 to TO005 pins taking into consideration the dead-time width (in case of interrupt culling ratio of 1/1).

[Output waveform width in respect to set value]

- PWM cycle = $BFCM03 \times 2 \times T_{TM00}$
- Dead-time width $T_{D0m} = (DTRR0 + 1)/f_{CLK}$
- Active width of positive phase (TO000, TO002, TO004 pins)
= $\{ (CM003 - CM00X_{up}) + (CM003 - CM00X_{down}) \} \times T_{TM00} - T_{D0m}$
- Active width of negative phase (TO001, TO003, TO005 pins)
= $(CM00X_{down} + CM00X_{up}) \times T_{TM00} - T_{D0m}$
- In this mode, $CM00X_{up} = CM00X_{down}$ (However, within the same PWM cycle).
Since $CM00X_{up}$ and $CM00X_{down}$ in the negative phase formula are prepared in a separate PWM cycle, $CM00X_{up} \neq CM00X_{down}$.

f_{CLK} :	Base clock
T_{TM00} :	TM00 count clock
$CM00X_{up}$:	Set value of CM000 to CM002 while TM00 is counting up
$CM00X_{down}$:	Set value of CM000 to CM002 while TM00 is counting down

The pin level when the TO000 to TO005 pins are reset is the high impedance state. When the control mode is selected thereafter, the following levels are output until the TM00 is started.

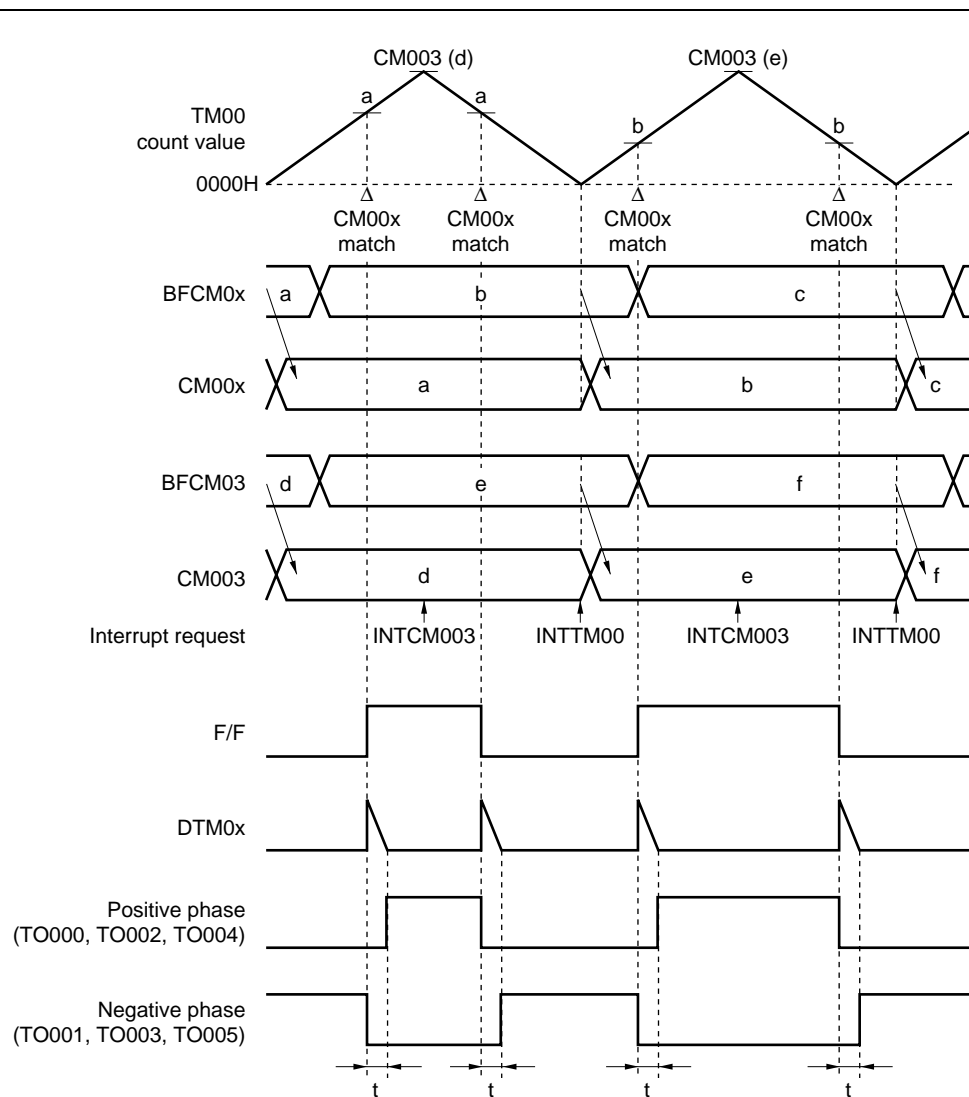
- TO000, TO002, TO004... When low active → High level
When high active → Low level
- TO001, TO003, TO005... When low active → Low level
When high active → High level

The active level is set with the ALVTO bit of the TOMR0 register. The default is low active.

Caution If a value such that the positive phase or negative phase active width is “0” or a negative value in the above formula, the TO000 to TO005 pins output a waveform fixed to the inactive level waveform with active width “0”.

Remark $m = 0$ to 2

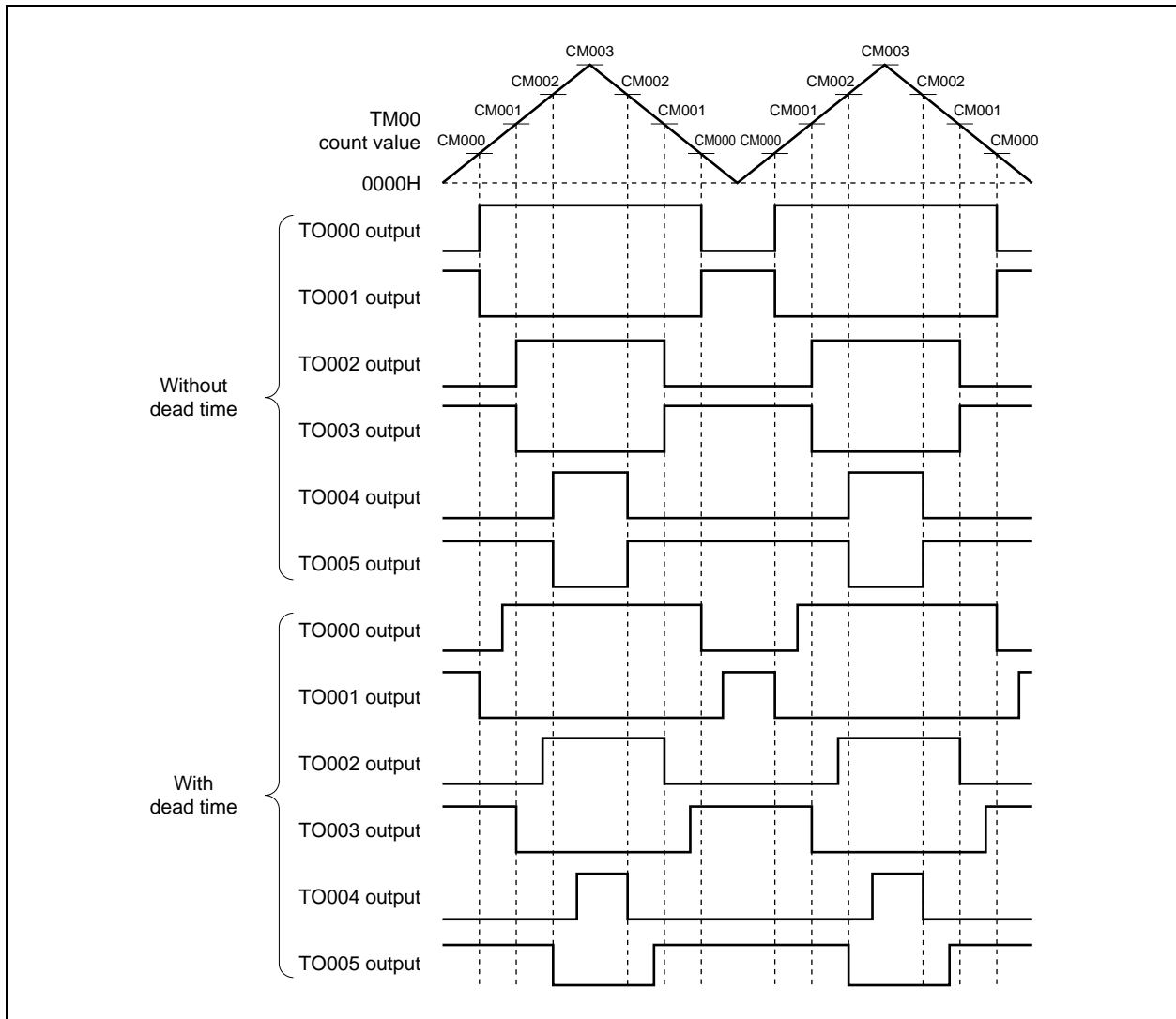
Figure 5-3. Operation Timing in PWM Mode 0 (Symmetric Triangular Wave)



- Remarks 1.** The above figure shows the timing chart when BFTE3 and BFTEN of the TMC00 register are 1, and transfers from BFCM03 to CM003, and from BFCM0 to CM00x are enabled. Transfer is not performed when BFTE3 = 0, BFTEN = 0.
2. x = 0 to 2
 3. t : Dead time = $(DTRR0 + 1)/f_{CLK}$ (f_{CLK} : Base clock)
 4. To not use dead time, set the TMC0CED0 bit of the TMC00 register to 1.
 5. The above figure shows an active high case.

Figure 5-4 shows the overall operation image.

Figure 5-4. Overall Operation Image of PWM Mode 0 (Symmetric Triangular Wave)



5.1.3 Register settings

(1) Timer 0 clock selection register (PRM01) settings

In this application circuit example, the PRM01 register is set as follows.

Caution Always set this register before using the timer.

```
PRM01 = 0x00;    /*fclk = fxx/2*/
```

In the case of the V850E/IA2, the PRM01 register should be set as follows.

```
PRM01 = 0x01;    /*fclk = fxx*/
```

Figure 5-5. Timer 0 Clock Selection Register (PRM01)

	7	6	5	4	3	2	1	0	Address	After reset
PRM01	0	0	0	0	0	0	0	PRM1	FFFFF5D0H	00H

Bit Position	Bit Name	Function
0	PRM1	Specifies the base clock (f_{CLK}) of timer 00 (TM00). 0: $f_{xx}/2$ (When $0 > 40$ MHz) 1: f_{xx} (When $0 \leq 40$ MHz) Remark f_{xx} : Internal system clock

(2) TOMR write enable register 0 (SPEC0) settings

In this application circuit example, the SPEC0 register is set as follows.

```
SPEC0 = 0x0000;    /*TOMR0 write enable*/
```

Figure 5-6. TOMR Write Enable Register 0 (SPEC0)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Address	After reset
SPEC0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	FFFFF580H	0000H

(3) Timer output mode register 0 (TOMR0) settings

In this application circuit example, the TOMR0 register is set as follows.

```
TOMR0 = 0x03;          /*output mode setting*/
```

Figure 5-7. Timer Output Mode Register 0 (TOMR0) (1/2)

	7	6	5	4	3	2	1	0	Address	After reset
TOMR0	ALVTO	ALVUB	ALVVB	ALVWB	TOSP	0	TOEDG1	TOEDG0	FFFFFF57DH	00H

Bit Position	Bit Name	Function
7	ALVTO	<p>Specifies the active level of the TO000, TO002, and TO004 pins.</p> <p>0: Active level is low level 1: Active level is high level</p> <p>Caution Changing the ALVTO bit during TM00 operation (TM0CE0 = 1) is prohibited.</p>
6	ALVUB	<p>Specifies the active level of the TO001 pin.</p> <p>0: Inverted level of active level set by ALVTO bit 1: Active level set by ALVTO bit</p> <p>When the ALVUB bit is 1, the TO001 output active level is the same level as TO000.</p> <p>Caution Changing the ALVUB bit during TM00 operation (TM0CE0 = 1) is prohibited.</p>
5	ALVVB	<p>Specifies the active level of the TO003 pin.</p> <p>0: Inverted level of active level set by ALVTO bit 1: Active level set by ALVTO bit</p> <p>When the ALVVB bit is 1, the TO003 output active level is the same level as TO002.</p> <p>Caution Changing the ALVVB bit during TM00 operation (TM0CE0 = 1) is prohibited.</p>
4	ALVWB	<p>Specifies the active level of the TO005 pin.</p> <p>0: Inverted level of active level set by ALVTO bit 1: Active level set by ALVTO bit</p> <p>When the ALVWB bit is 1, the TO005 output active level is the same level as TO004.</p> <p>Caution Changing the ALVWB bit during TM00 operation (TM0CE0 = 1) is prohibited.</p>
3	TOSP	<p>Controls TO000 to TO005 pin output stop through ESO0 pin input.</p> <p>0: Enables ESO0 pin input 1: Disables ESO0 pin input</p> <p>Cautions</p> <ol style="list-style-type: none"> 1. The output stop status can be released by writing "1" to the TORS0 bit of the TUC00 register. The operation continues even if output is prohibited for all timers and counters. 2. Before changing the ESO0 pin input status from disable to enable (changing TOSP bit from 1 to 0), write 1 to the TORS0 bit of the TUC0 register to reset the ESO0 pin input status.

Figure 5-7. Timer Output Mode Register 0 (TOMR0) (2/2)

Bit Position	Bit Name	Function															
1, 0	TOEDG1, TOEDG0	<p>These bits select the valid edge or level when setting forcible stop of TO000 to TO005 output through ESO0 pin input with the TOSP bit.</p> <table border="1"> <thead> <tr> <th>TOEDG1</th> <th>TOEDG0</th> <th>Operation</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Rising edge</td> </tr> <tr> <td>0</td> <td>1</td> <td>Falling edge</td> </tr> <tr> <td>1</td> <td>0</td> <td>Low level</td> </tr> <tr> <td>1</td> <td>1</td> <td>High level</td> </tr> </tbody> </table> <p>Cautions</p> <ol style="list-style-type: none"> 1. Changing the TOEDG1 and TOEDG0 bits during TM00 operation (TM0CE0 = 1) is prohibited. 2. Before changing the settings of the TOEDG1 and TOEDG0 bits, write 1 to the TORS0 bit of the TUC00 register to reset the ESO0 pin input status. 	TOEDG1	TOEDG0	Operation	0	0	Rising edge	0	1	Falling edge	1	0	Low level	1	1	High level
TOEDG1	TOEDG0	Operation															
0	0	Rising edge															
0	1	Falling edge															
1	0	Low level															
1	1	High level															

Data setting to timer output mode register 0 (TOMR0) is done in the following sequence.

- <1> Prepare the data to be set to timer output mode register 0 (TOMR0) in a general-purpose register.
- <2> Write data to TOMR write enable register 0 (SEPC0).
- <3> Set timer output mode register 0 (TOMR0) (performed with the following instructions).
 - Store instruction (ST/SST instructions)
 - Bit manipulation instruction (SET1/CLR1/NOT1 instructions)

[Description Example]

```

<1> MOV    0x04, r10
<2> ST.B   r10, SPEC0 [r0]
<3> ST.B   r10, TOMR0 [r0]

```

To read the TOMR0 register, no special sequence is required.

- Cautions**
1. Prohibit interrupts between SPEC0 issuance (<2>) and TOMR0 register write that immediately follows (<3>).
 2. The data written to the SPEC0 register is dummy data; use the same register as the general-purpose register used to set the TOMR0 register (<3> in the above example) for SPEC0 register write (<2> in the above example). The same applies when using a general-purpose register for addressing.
 3. Do not write to the SPEC0 register or TOMR0 register via DMA transfer.

(4) PWM software timing output register 0 (PSTO0)

In this application circuit example, the PSTO0 register is set as follows.

```
PSTO0 = 0x00;          /*real-time output prohibited*/
```



Figure 5-8. PWM Software Timing Output Register 0 (PSTO0) (1/2)

	<7>	6	5	4	3	<2>	<1>	<0>	Address	After reset
PSTO0	TORTO0	0	0	0	0	U	V	W	FFFFFF57EH	00H

Bit Position	Bit Name	Function																								
7	TORTO0	Specifies TO000 to TO005 output control. 0: Timer output 1: Software output The change of the TO000 to TO005 signals during software output occurs when the TORTO0 bit is set (to 1) and a value is written to the UPORT0, VPORT0, and WPORT0 bits. A dead-time timer can also be used.																								
2	U	Specifies the TO000 (U phase)/TO001 (\bar{U} phase) pin output value. <table border="1" style="width:100%; border-collapse: collapse; margin-top: 5px;"> <thead> <tr> <th style="width:10%;">U</th> <th colspan="3">Operation</th> </tr> </thead> <tbody> <tr> <td style="text-align:center">0</td> <td style="width:10%;">TO000</td> <td colspan="2">Inverted level of ALVTO bit setting</td> </tr> <tr> <td rowspan="2"></td> <td rowspan="2" style="text-align:center">TO001</td> <td>When ALVUB = 0</td> <td>Level of ALVTO bit setting</td> </tr> <tr> <td>When ALVUB = 1</td> <td>Inverted level of ALVTO bit setting</td> </tr> <tr> <td style="text-align:center">1</td> <td style="text-align:center">TO000</td> <td colspan="2">Level of ALVTO bit setting</td> </tr> <tr> <td rowspan="2"></td> <td rowspan="2" style="text-align:center">TO001</td> <td>When ALVUB = 0</td> <td>Inverted level of ALVTO bit setting</td> </tr> <tr> <td>When ALVUB = 1</td> <td>Level of ALVTO bit setting</td> </tr> </tbody> </table> <p>Caution If the UPORT0 bit setting value is changed when TORTO0 = 1, the dead-time setting becomes valid for the TO000/TO001 output signal in the same way as during normal timer operation.</p>	U	Operation			0	TO000	Inverted level of ALVTO bit setting			TO001	When ALVUB = 0	Level of ALVTO bit setting	When ALVUB = 1	Inverted level of ALVTO bit setting	1	TO000	Level of ALVTO bit setting			TO001	When ALVUB = 0	Inverted level of ALVTO bit setting	When ALVUB = 1	Level of ALVTO bit setting
U	Operation																									
0	TO000	Inverted level of ALVTO bit setting																								
	TO001	When ALVUB = 0	Level of ALVTO bit setting																							
		When ALVUB = 1	Inverted level of ALVTO bit setting																							
1	TO000	Level of ALVTO bit setting																								
	TO001	When ALVUB = 0	Inverted level of ALVTO bit setting																							
		When ALVUB = 1	Level of ALVTO bit setting																							
1	V	Specifies the TO002 (V phase)/TO003 (\bar{V} phase) pin output value. <table border="1" style="width:100%; border-collapse: collapse; margin-top: 5px;"> <thead> <tr> <th style="width:10%;">V</th> <th colspan="3">Operation</th> </tr> </thead> <tbody> <tr> <td style="text-align:center">0</td> <td style="width:10%;">TO002</td> <td colspan="2">Inverted level of ALVTO bit setting</td> </tr> <tr> <td rowspan="2"></td> <td rowspan="2" style="text-align:center">TO003</td> <td>When ALVVB = 0</td> <td>Level of ALVTO bit setting</td> </tr> <tr> <td>When ALVVB = 1</td> <td>Inverted level of ALVTO bit setting</td> </tr> <tr> <td style="text-align:center">1</td> <td style="text-align:center">TO002</td> <td colspan="2">Level of ALVTO bit setting</td> </tr> <tr> <td rowspan="2"></td> <td rowspan="2" style="text-align:center">TO003</td> <td>When ALVVB = 0</td> <td>Inverted level of ALVTO bit setting</td> </tr> <tr> <td>When ALVVB = 1</td> <td>Level of ALVTO bit setting</td> </tr> </tbody> </table> <p>Caution If the VPORT0 bit setting value is changed when TORTO0 = 1, the dead-time setting becomes valid for the TO002/TO003 output signal in the same way as during normal timer operation.</p>	V	Operation			0	TO002	Inverted level of ALVTO bit setting			TO003	When ALVVB = 0	Level of ALVTO bit setting	When ALVVB = 1	Inverted level of ALVTO bit setting	1	TO002	Level of ALVTO bit setting			TO003	When ALVVB = 0	Inverted level of ALVTO bit setting	When ALVVB = 1	Level of ALVTO bit setting
V	Operation																									
0	TO002	Inverted level of ALVTO bit setting																								
	TO003	When ALVVB = 0	Level of ALVTO bit setting																							
		When ALVVB = 1	Inverted level of ALVTO bit setting																							
1	TO002	Level of ALVTO bit setting																								
	TO003	When ALVVB = 0	Inverted level of ALVTO bit setting																							
		When ALVVB = 1	Level of ALVTO bit setting																							

Remark ALVTO bit: Bit 7 of the TOMR0 register
 ALVUB bit: Bit 6 of the TOMR0 register
 ALVVB bit: Bit 5 of the TOMR0 register

Figure 5-8. PWM Software Timing Output Register 0 (PSTO0) (2/2)

Bit Position	Bit Name	Function																					
0	WPORT0	Specifies the TO004 (W phase)/TO005 (\bar{W} phase) pin output value.																					
		<table border="1"> <thead> <tr> <th>WPORT0</th> <th colspan="2">Operation</th> </tr> </thead> <tbody> <tr> <td rowspan="2">0</td> <td>TO004</td> <td>Inverted level of ALVTO bit setting</td> </tr> <tr> <td rowspan="2">TO005</td> <td>When ALVWB = 0</td> <td>Level of ALVTO bit setting</td> </tr> <tr> <td></td> <td>When ALVWB = 1</td> <td>Inverted level of ALVTO bit setting</td> </tr> <tr> <td rowspan="2">1</td> <td>TO004</td> <td>Inverted level of ALVTO bit setting</td> </tr> <tr> <td rowspan="2">TO005</td> <td>When ALVWB = 0</td> <td>Inverted level of ALVTO bit setting</td> </tr> <tr> <td></td> <td>When ALVWB = 1</td> <td>Level of ALVTO bit setting</td> </tr> </tbody> </table>	WPORT0	Operation		0	TO004	Inverted level of ALVTO bit setting	TO005	When ALVWB = 0	Level of ALVTO bit setting		When ALVWB = 1	Inverted level of ALVTO bit setting	1	TO004	Inverted level of ALVTO bit setting	TO005	When ALVWB = 0	Inverted level of ALVTO bit setting		When ALVWB = 1	Level of ALVTO bit setting
WPORT0	Operation																						
0	TO004	Inverted level of ALVTO bit setting																					
	TO005	When ALVWB = 0	Level of ALVTO bit setting																				
		When ALVWB = 1	Inverted level of ALVTO bit setting																				
1	TO004	Inverted level of ALVTO bit setting																					
	TO005	When ALVWB = 0	Inverted level of ALVTO bit setting																				
		When ALVWB = 1	Level of ALVTO bit setting																				
		Caution If the WPORT0 bit setting value is changed when TORTO0 = 1, the dead-time setting becomes valid for the TO004/TO005 output signal in the same way as during normal timer operation.																					
Remark n = 0, 1 ALVTO bit: Bit 7 of the TOMR0 register ALVWB bit: Bit 4 of the TOMR0 register																							

★

The TO000 to TO005 pins can be set to timer output by a match between TM00 and the compare register or to software output using the PSTO0 register (TORTO0 bit = 1). Software output has priority over timer output.

Consequently, when the setting changes from TM0CE0 = 1 (timer operation enabled), TORTO0 = 1 (software output enabled) to TM0CE0 = 1 (timer operation enabled), TORTO0 = 0 (software output disabled), the TO000 to TO005 pins continue to perform software output until the occurrence of the first F/F set/reset due to a match between TM0n and the compare register after the TORTO0 bit setting changes.

The relationship between the settings of the TORTO0 and TM0CEn bits when ALVTO = 1 and the output of TO000 (negative phase side) is shown on the following pages (the positive phase side (TO001, TO003, and TO005) is dependent on the ALVUB, ALVVB, and ALVWB bits, so refer to the explanations of each of these bits).

(5) Buffer registers CM00 to CM03 (BFCM00 to BFCM03) settings

In this application circuit example, the BFCM00 to BFCM03 registers are set as shown below.

```
BFCM00 = 312;          /*Initial value 50% duty*/
BFCM01 = 312;          /*Initial value 50% duty*/
BFCM02 = 312;          /*Initial value 50% duty*/
BFCM03 = 625;          /*20 kHz*/
```

★

In the case of the V850E/IA2, the BFCM00 to BFCM03 registers are set as shown below.

```
BFCM00 = 500;          /*Initial value 50% duty*/
BFCM01 = 500;          /*Initial value 50% duty*/
BFCM02 = 500;          /*Initial value 50% duty*/
BFCM03 = 1000;        /*20 kHz*/
```

The transfer operation from the BFCM00 to BFCM03 registers to the CM00 to CM03 registers is performed during the operation of TM00 (TM0CE0 bit of TMC00 register = 1), so it is performed when an underflow interrupt (INTTM00) occurs.

Figure 5-9. Buffer Registers CM00 to CM03 (BFCM00 to BFCM03)

BFCM00	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Address	After reset
	[16-bit register box]																FFFFF572H	FFFFH
BFCM01	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Address	After reset
	[16-bit register box]																FFFFF574H	FFFFH
BFCM02	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Address	After reset
	[16-bit register box]																FFFFF576H	FFFFH
BFCM03	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Address	After reset
	[16-bit register box]																FFFFF578H	FFFFH

(6) Dead-time timer reload register 0 (DTRR0) settings

In this application circuit example, the DTRR0 register is set as shown below.

```
DTRR0 = 50;          /*Dead time 2 us*/
```

★

In the case of the V850E/IA2, the DTRR0 register is set as shown below.

```
DTRR0 = 80;          /*Dead time 2 us*/
```

Figure 5-10. Dead-time timer Reload Register 0 (DTRR0)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Address	After reset
DTRR0	0	0	0	0													FFFFFF570H	FFFH

(7) PWM output enable register 0 (POER0)

In this application circuit example, the POER0 register is set as follows.

```
POER0 = 0x3f;          /*all phases active*/
```

When an error occurs, output is disabled by writing 00H to the POER0 register.

Figure 5-11. PWM Output Enable Register 0 (POER0)

Bit Position	Bit Name	Function
5	OE210	Specifies the output status of the TO005 pin. 0: TO005 output status is high impedance. 1: TO005 output status is controlled by TM0CE0 bit of TMC00 register and TORTO0 bit of PSTO0 register and ESO0 pin.
4	OE200	Specifies the output status of the TO004 pin. 0: TO004 output status is high impedance. 1: TO004 output status is controlled by TM0CE0 bit of TMC00 register and TORTO0 bit of PSTO0 register and ESO0 pin.
3	OE110	Specifies the output status of the TO003 pin. 0: TO003 output status is high impedance. 1: TO003 output status is controlled by TM0CE0 bit of TMC00 register and TORTO0 bit of PSTO0 register and ESO0 pin.
2	OE100	Specifies the output status of the TO002 pin. 0: TO002 output status is high impedance. 1: TO002 output status is controlled by TM0CE0 bit of TMC00 register and TORTO0 bit of PSTO0 register and ESO0 pin.
1	OE010	Specifies the output status of the TO001 pin. 0: TO001 output status is high impedance. 1: TO001 output status is controlled by TM0CE0 bit of TMC00 register and TORTO0 bit of PSTO0 register and ESO0 pin.
0	OE000	Specifies the output status of the TO000 pin. 0: TO000 output status is high impedance. 1: TO000 output status is controlled by TM0CE0 bit of TMC00 register and TORTO0 bit of PSTO0 register and ESO0 pin.

POER0	7	6	<5>	<4>	<3>	<2>	<1>	<0>	Address	After reset
	0	0	OE210	OE200	OE110	OE100	OE010	OE000	FFFFFF57FH	00H

(8) Timer control register 00 (TMC00) settings

In this application circuit example, the TMC00 register is set as follows.

```
TMC00 = 0x8018;          /*TM00 timer start*/
```

Figure 5-12. Timer Control Register 00 (TMC00) (1/3)

	<15>	<14>	13	12	11	10	9	8	7	6	<5>	4	3	2	1	0	Address	After reset
TMC00	TMOCE0	STINT0	CUL02	CUL01	CUL00	PRM02	PRM01	PRM00	0	0	TMOCE0	BFTE3	BFTEN	MBFTE	MOD01	MOD00	FFFFFF57AH	0508H

Bit Position	Bit Name	Function																												
15	TMOCE0	Specifies the operation of TM00. 0: Count disabled (stops after all count values are cleared) 1: Count enabled Caution When TMOCE0 = 0, TO000 to TO005 outputs become high impedance.																												
14	STINT0	Specifies interrupt occurrence during TM00 timer start. 0: Interrupt does not occur at operation start 1: Interrupt occurs at operation start When STINT0 bit = 1, an interrupt occurs immediately after the rising edge of the TMOCE0 signal. When MOD01 bit = 0 (triangular wave mode), the INTTM00 interrupt occurs, and when the MOD01 bit = 1 (sawtooth wave mode), the INTCM003 interrupt occurs. Caution Changing the STINT0 bit during TM00 operation (TMOCE0 bit = 1) is prohibited.																												
13 to 11	CUL02 to CUL00	Specifies the interrupt culling ratio. <table border="1" style="width:100%; border-collapse: collapse; margin-top: 10px;"> <thead> <tr> <th style="width:10%;">CUL02</th> <th style="width:10%;">CUL01</th> <th style="width:10%;">CUL00</th> <th style="width:70%;">Interrupt Culling Ratio</th> </tr> </thead> <tbody> <tr> <td style="text-align:center">0</td> <td style="text-align:center">0</td> <td style="text-align:center">0</td> <td style="text-align:center">1/1</td> </tr> <tr> <td style="text-align:center">0</td> <td style="text-align:center">0</td> <td style="text-align:center">1</td> <td style="text-align:center">1/2</td> </tr> <tr> <td style="text-align:center">0</td> <td style="text-align:center">1</td> <td style="text-align:center">0</td> <td style="text-align:center">1/4</td> </tr> <tr> <td style="text-align:center">0</td> <td style="text-align:center">1</td> <td style="text-align:center">1</td> <td style="text-align:center">1/8</td> </tr> <tr> <td style="text-align:center">1</td> <td style="text-align:center">0</td> <td style="text-align:center">0</td> <td style="text-align:center">1/16</td> </tr> <tr> <td colspan="3">Other than above</td> <td style="text-align:center">Culling not performed</td> </tr> </tbody> </table> Cautions <ol style="list-style-type: none"> 1. The INTTM00 interrupt and INTCM003 interrupt can be culled with the same culling ratio (1/1, 1/2, 1/4, 1/8, 1/16). 2. Even when BFTE3 bit = 1, BFTEN bit = 1 (settings to transfer data from BFCM00 to BFCM03 registers to CM000 to CM003 registers), transfer is not performed with the generation timing of culled INTTM00 and INTCM003 interrupts if MBFTE bit = 0. 3. If the culling ratio is changed during a count operation, the new culling ratio is applied after an interrupt has occurred with the culling ratio prior to the change. 	CUL02	CUL01	CUL00	Interrupt Culling Ratio	0	0	0	1/1	0	0	1	1/2	0	1	0	1/4	0	1	1	1/8	1	0	0	1/16	Other than above			Culling not performed
CUL02	CUL01	CUL00	Interrupt Culling Ratio																											
0	0	0	1/1																											
0	0	1	1/2																											
0	1	0	1/4																											
0	1	1	1/8																											
1	0	0	1/16																											
Other than above			Culling not performed																											

Figure 5-12. Timer Control Register 00 (TMC00) (2/3)

Bit Position	Bit Name	Function																																
10 to 8	PRM02 to PRM00	<p>Specifies the count clock for TM00.</p> <table border="1"> <thead> <tr> <th>PRM02</th> <th>PRM01</th> <th>PRM00</th> <th>Count Clock</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>f_{CLK}</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>f_{CLK}/2</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>f_{CLK}/4</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>f_{CLK}/8</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>f_{CLK}/16</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>f_{CLK}/32</td> </tr> <tr> <td colspan="3">Other than above</td> <td>Setting prohibited</td> </tr> </tbody> </table> <p>Caution The divide ratio switch timing is after the TM00 value has become 0000H and an INTTM00 interrupt has occurred. Therefore, in the timing that corresponds to interrupt culling, the divide ratio is not switched.</p> <p>Remark For the base clock (f_{CLK}), see 5.1.3 (1) Timer 0 clock selection register (PRM01).</p>	PRM02	PRM01	PRM00	Count Clock	0	0	0	f _{CLK}	0	0	1	f _{CLK} /2	0	1	0	f _{CLK} /4	0	1	1	f _{CLK} /8	1	0	0	f _{CLK} /16	1	0	1	f _{CLK} /32	Other than above			Setting prohibited
PRM02	PRM01	PRM00	Count Clock																															
0	0	0	f _{CLK}																															
0	0	1	f _{CLK} /2																															
0	1	0	f _{CLK} /4																															
0	1	1	f _{CLK} /8																															
1	0	0	f _{CLK} /16																															
1	0	1	f _{CLK} /32																															
Other than above			Setting prohibited																															
5	TM0CED0	<p>Specifies the operation of the DTM00 to DTM02 timers.</p> <p>0: DTM00 to DTM02 perform count operation 1: DTM00 to DTM02 stopped</p> <p>Cautions</p> <ol style="list-style-type: none"> 1. Changing the TM0CED0 bit during TM00 operation (TM0CE0 = 1) is prohibited. 2. If TM00 is operated when TM0CED0 bit = 1, a signal without dead time is output to the TO000 to TO005 pins. 																																
4	BFTE3	<p>Specifies transfer of data from the BFCM03 register to the CM003 register.</p> <p>0: Transfer disabled 1: Transfer enabled</p> <p>The transfer timing from the BFCM03 register to the CM003 register is as follows.</p> <table border="1"> <thead> <tr> <th>BFTE3</th> <th>TM00 Operation Mode</th> <th>BFCM03 → CM003 Transfer Timing</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>All modes</td> <td>No transfer</td> </tr> <tr> <td>1</td> <td>PWM mode 0 (symmetric triangular wave)</td> <td>INTTM00</td> </tr> </tbody> </table> <p>When BFTE3 bit = 1, the value of the BFCM03 register is transferred to the CM003 register upon occurrence of an INTTM00 or INTCM003 interrupt.</p>	BFTE3	TM00 Operation Mode	BFCM03 → CM003 Transfer Timing	0	All modes	No transfer	1	PWM mode 0 (symmetric triangular wave)	INTTM00																							
BFTE3	TM00 Operation Mode	BFCM03 → CM003 Transfer Timing																																
0	All modes	No transfer																																
1	PWM mode 0 (symmetric triangular wave)	INTTM00																																

Figure 5-12. Timer Control Register 00 (TMC00) (3/3)

Bit Position	Bit Name	Function																						
3	BFTEN	<p>Specifies transfer of data from the BFCM00 to BFCM02 registers to the CM000 to CM002 registers.</p> <p>0: Transfer disabled 1: Transfer enabled</p> <table border="1"> <thead> <tr> <th>BFTEN</th> <th>TM00 Operation Mode</th> <th>BFCM00 to BFCM02 → CM000 to CM002 Transfer Timing</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>All modes</td> <td>No transfer</td> </tr> <tr> <td>1</td> <td>PWM mode 0 (symmetric triangular wave)</td> <td>INTTM00</td> </tr> </tbody> </table> <p>When the BFTEN bit = 1, the values of the BFCM00 to BFCM02 registers are transferred to the CM000 to CM002 registers upon occurrence of an INTTM00 or INTCM003 interrupt.</p>	BFTEN	TM00 Operation Mode	BFCM00 to BFCM02 → CM000 to CM002 Transfer Timing	0	All modes	No transfer	1	PWM mode 0 (symmetric triangular wave)	INTTM00													
BFTEN	TM00 Operation Mode	BFCM00 to BFCM02 → CM000 to CM002 Transfer Timing																						
0	All modes	No transfer																						
1	PWM mode 0 (symmetric triangular wave)	INTTM00																						
2	MBFTE	<p>When culling of the INTTM00 and INTCM003 interrupts is set with the CUL02 to CUL00 bits, specifies whether enable or disable the BFTE3 and BFTEN bit settings upon occurrence of an interrupt for culling.</p> <p>0: Disable the set values of BFTE3, BFTEN bits upon occurrence of a culling interrupt 1: Enable the set values of BFTE3, BFTEN bits upon occurrence of a culling interrupt</p> <p>The various combinations are as follows.</p> <table border="1"> <thead> <tr> <th colspan="2">MBFTE</th> <th colspan="2">Operation upon Occurrence of Interrupt for Culling</th> </tr> <tr> <th colspan="2"></th> <th>0</th> <th>1</th> </tr> </thead> <tbody> <tr> <td rowspan="2">BFTEN</td> <td>0</td> <td>BFCM00 to BFCM02 → CM000 to CM002 transfer disabled</td> <td>BFCM00 to BFCM02 → CM000 to CM002 transfer disabled</td> </tr> <tr> <td>1</td> <td>BFCM00 to BFCM02 → CM000 to CM002 transfer disabled</td> <td>BFCM00 to BFCM02 → CM000 to CM002 transfer enabled</td> </tr> <tr> <td rowspan="2">BFTE3</td> <td>0</td> <td>BFCM03 → CM003 transfer disabled</td> <td>BFCM03 → CM003 transfer disabled</td> </tr> <tr> <td>1</td> <td>BFCM03 → CM003 transfer disabled</td> <td>BFCM03 → CM003 transfer enabled</td> </tr> </tbody> </table>	MBFTE		Operation upon Occurrence of Interrupt for Culling				0	1	BFTEN	0	BFCM00 to BFCM02 → CM000 to CM002 transfer disabled	BFCM00 to BFCM02 → CM000 to CM002 transfer disabled	1	BFCM00 to BFCM02 → CM000 to CM002 transfer disabled	BFCM00 to BFCM02 → CM000 to CM002 transfer enabled	BFTE3	0	BFCM03 → CM003 transfer disabled	BFCM03 → CM003 transfer disabled	1	BFCM03 → CM003 transfer disabled	BFCM03 → CM003 transfer enabled
MBFTE		Operation upon Occurrence of Interrupt for Culling																						
		0	1																					
BFTEN	0	BFCM00 to BFCM02 → CM000 to CM002 transfer disabled	BFCM00 to BFCM02 → CM000 to CM002 transfer disabled																					
	1	BFCM00 to BFCM02 → CM000 to CM002 transfer disabled	BFCM00 to BFCM02 → CM000 to CM002 transfer enabled																					
BFTE3	0	BFCM03 → CM003 transfer disabled	BFCM03 → CM003 transfer disabled																					
	1	BFCM03 → CM003 transfer disabled	BFCM03 → CM003 transfer enabled																					
1, 0	MOD01, MOD00	<p>Specifies the operation mode of TM00.</p> <table border="1"> <thead> <tr> <th>MOD01</th> <th>MOD00</th> <th>Operation Mode</th> <th>TM00 Operation</th> <th>Timer Clear Cause</th> <th>BFCM03 → CM003 Timing</th> <th>BFCM00 to BFCM02 → CM000 to CM002 Timing</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>PWM mode 0 (symmetric triangular wave)</td> <td>Up/down</td> <td>–</td> <td>INTTM00</td> <td>INTTM00</td> </tr> <tr> <td colspan="2">Other than above</td> <td colspan="5">Setting prohibited</td> </tr> </tbody> </table> <p>Caution Changing the value of the MOD01, MOD00 bits during TM00 operation (TMOCE0 bit = 1) is prohibited.</p>	MOD01	MOD00	Operation Mode	TM00 Operation	Timer Clear Cause	BFCM03 → CM003 Timing	BFCM00 to BFCM02 → CM000 to CM002 Timing	0	0	PWM mode 0 (symmetric triangular wave)	Up/down	–	INTTM00	INTTM00	Other than above		Setting prohibited					
MOD01	MOD00	Operation Mode	TM00 Operation	Timer Clear Cause	BFCM03 → CM003 Timing	BFCM00 to BFCM02 → CM000 to CM002 Timing																		
0	0	PWM mode 0 (symmetric triangular wave)	Up/down	–	INTTM00	INTTM00																		
Other than above		Setting prohibited																						

(9) Timer unit control register 00 (TUC00) settings

In this circuit example, the TUC00 register is set as shown below.

Figure 5-13. Timer Unit Control Register 00 (TUC00)

	7	6	5	4	3	2	<1>	<0>	Address	After reset
TUC00	0	0	0	0	0	0	TORS0	TOSTA0	FFFFFF57CH	01H

Bit Position	Bit Name	Function
1	TORS0	<p>Flag that restarts the output of the TO000 to TO005 pins, which was forcibly stopped through ESO0 pin input. Output is resumed by writing "1" to the TORS0 bit.</p> <p>Cautions</p> <ol style="list-style-type: none"> 1. If the level is the ESO0 pin input level setting (TOMR register TOEDG1 bit = 1, TOEDG0 bit = 0 or 1), the output disabled state is not released (TOSTA0 bit = 1) even if "1" is written to the TORS0 bit while the level is the output disable state (TOSTA0 bit = 0). The output disabled state is released when the input level becomes the inactive level (TOSTA0 bit = 0). 2. When the ESO0 pin input is the edge setting (TOEDG1 bit = 0, TOEDG0 bit = 0 or 1), the output disabled state (TOSTA0 bit = 1) is released (TOSTA0 bit = 0) by writing 1 to the TORS0 bit. 3. After reset, be sure to write "1" to the TORS0 bit prior to starting TO000 to TO005 pin output. A "0" is read when the TORS0 bit is read.
0	TOSTA0	<p>TO000 to TO005 pin output status flag through ESO0 pin input</p> <p>0: Output enabled status 1: Output disabled status</p>

★

5.2 Encoder Counter Functions (Timer 10 [TM10])

★ The V850E/IA1 includes a 2-channel (1 channel for the V850E/IA2) 16-bit up/down counter that can be used as a 2-phase encoder input or general-purpose timer. In this application circuit example, timer 10 (TM10) uses one of these channels as a timer for 2-phase encoder input.

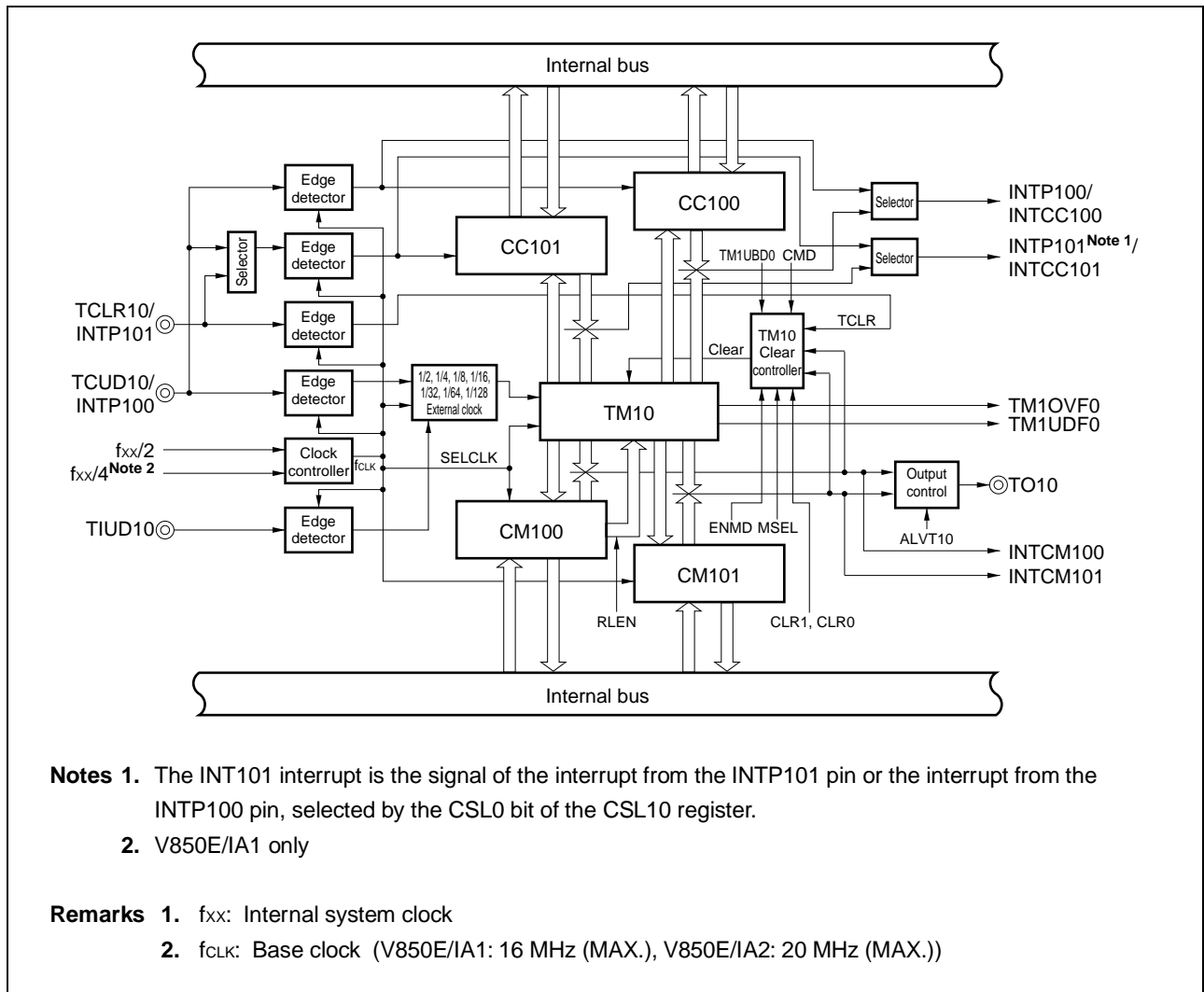
The following settings are made in the application circuit example.

- UDC mode A
- Counter is cleared at the falling edge of TCLR10
- Uses $\times 4$ frequency multiplication (mode 4)

5.2.1 General

This section shows block diagrams and provides general descriptions of registers.

★ **Figure 5-14. Block Diagram of Timer 10 (TM10)**



(1) Timer 10 (TM10)

TM10 functions as a 2-phase encoder input up/down counter and general-purpose timer.

- Cautions**
1. Writing to TM10 is enabled only when the TM1CE0 bit of the TMC10 register is “0” (count operation disabled).
 2. It is prohibited to set the CMD bit (general-purpose timer mode) and the MSEL bit (UDC mode B) of the TUM0 register to “0” and “1”, respectively.
 3. Continuous reading of TM10 is prohibited. If TM10 is continuously read, the second read value may differ from the actual value. If TM10 must be read twice, be sure to read another register between the first and the second read operation.

Correct usage example

TM10 read
 TM11 read
 TM10 read
 TM11 read

Incorrect usage example

TM10 read
 TM10 read
 TM11 read
 TM11 read

TM10 start and stop is controlled by the TM1CE0 bit of timer control register 10 (TMC10).

The TM10 operation consists of two modes: the general-purpose timer mode and the up/down counter mode (UDC mode).

The up/down counter mode (UDC mode) is used in this application circuit example.

(a) Up/down counter mode (UDC mode)

In the UDC mode, TM10 functions as a 16-bit up/down counter, counting based on the TCUD10 and TIUD10 input signals.

Two operation modes, the UDC A and UDC B modes, can be set with the MSEL bit of the TUM0 register for this mode.

The UDC mode A is used in this application circuit example.

- UDC mode A (when CMD bit = 1, MSEL bit = 0)

TM10 can be cleared by setting the CLR1 and CLR0 bits of the TMC10 register.

When the TM1CE0 bit of the TMC10 register is “1”, TM10 counts up/down when the operation mode is the UDC mode.

The conditions for clearing the TM10 are classified as follows.

Table 5-1. Timer 1 (TM10) Clear Conditions

TMC10 Register		TM10 Clear
CLR1 Bit	CLR0 Bit	
0	0	Cleared only by TCLR10 input
0	1	Cleared upon match with CM100 set value during up count operation
1	0	Cleared by TCLR10 input or upon match with CM100 set value during up count operation
1	1	Clearing not performed

(2) Compare register 100 (CM100)

CM100 is a 16-bit register that always compares its value with the value of TM10. When the value of a compare register matches the value of TM10, an interrupt signal is generated. In UDC mode A (MSEL bit of TUM0 register = 0), an interrupt signal (INTCM100) is always generated upon occurrence of a match.

Caution When the TM1CE0 bit of the TMC10 register is “1”, it is prohibited to overwrite the value of the CM100 register.

(3) Compare register 101 (CM101)

CM101 is a 16-bit register that always compares its value with the value of TM10. When the value of a compare register matches the value of TM10, an interrupt signal is generated. In UDC mode A (MSEL bit of TUM0 register = 0), an interrupt signal (INTCM101) is always generated upon occurrence of a match.

Caution When the TM1CE0 bit of the TMC10 register is “1”, it is prohibited to overwrite the value of the CM101 register.

5.2.2 Use of encoder counter in application circuit example

Four types of count operation modes are available in UDC mode (when CMD bit of TUM10 register = 1). In this application circuit example, mode 4 is used when counting A-phase and B-phase encoders.

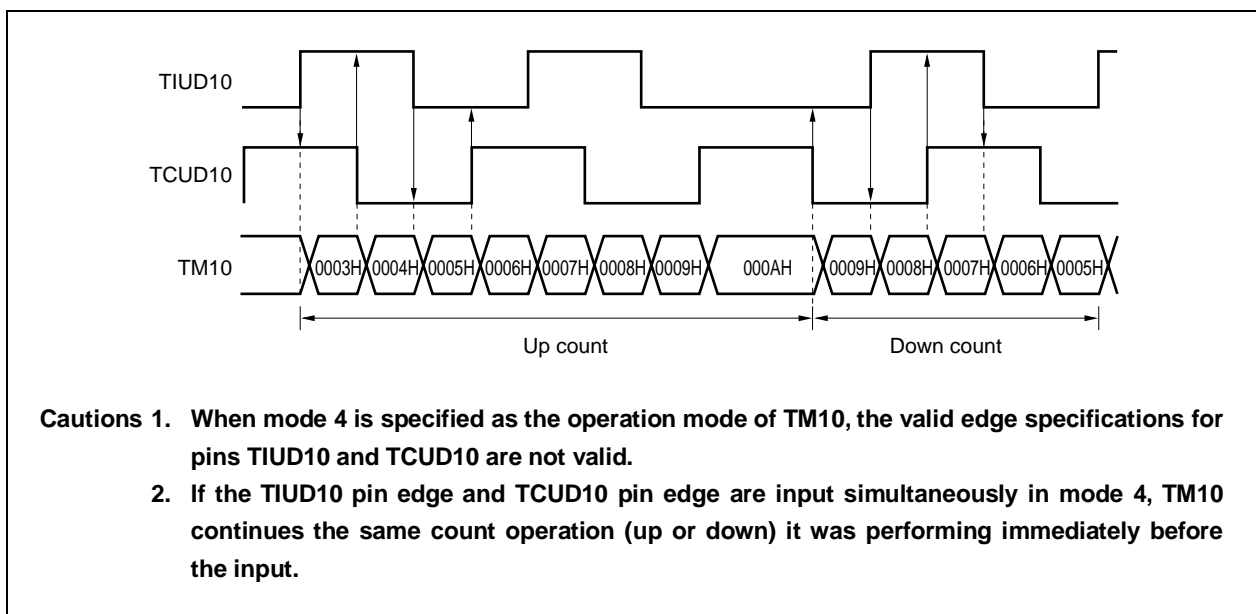
Mode 4 and operations in UDC mode A are described below.

(1) Mode 4 (PRM12 = 1, PRM11 = 1, PRM10 = 1)

In mode 4, when a 2-phase signal 90 degrees out of phase is input to the TIUD10 and TCUD10 pins, an up/down operation is automatically judged and counting is performed according to the timing shown in **Figure 5-15**.

In mode 4, counting is executed at both the rising and falling edges of the 2-phase signal input to the TIUD10 and TCUD10 pins. Therefore, TM10 counts four times per cycle of an input signal ($\times 4$ count).

Figure 5-15. Mode 4



Cautions 1. When mode 4 is specified as the operation mode of TM10, the valid edge specifications for pins TIUD10 and TCUD10 are not valid.

2. If the TIUD10 pin edge and TCUD10 pin edge are input simultaneously in mode 4, TM10 continues the same count operation (up or down) it was performing immediately before the input.

In this application circuit example, mode 4 is used, so that the 2,500 pulses per rotation is multiplied by four to become 10,000 pulses. To obtain the positions along the motor axes, the count value is cleared whenever TCLR10 (Z phase) is input.

(2) Operation in UDC mode A

In UDC mode A, the encoder counter operates as a 16-bit up/down counter.

The count clock input to TM10 in UDC mode A (CMD bit of TUM0 register = 1) can only be external input from the TIUD10 and TCUD10 pins. Up/down count judgment in UDC mode A is determined based on the phase difference of the TIUD10 and TCUD10 pin inputs according to the PRM10 register setting.

In mode 4, the up/down count is automatically judged by the detection of both edges of the TIUD10 and TCUD10 inputs.

In UDC mode A, the TM10 clear cause can be selected as only external clear input (TCLR10), a match signal between the TM10 count value and the CM100 set value during up count operation, or logical sum (OR) of the two signals, using bits CLR1 and CLR0 of the TMC10 register.

TM10 can transfer the value of CM100 upon occurrence of TM10 underflow.

5.2.3 Register settings

(1) Timer 1/timer 2 clock selection register (PRM02) settings

In this application circuit example, the PRM02 register is set as follows.

★

```
PRM02 = 0;          /*fclk = fxx /4*/
```

★

In the case of the V850E/IA2, the PRM02 register is set as shown below.

```
PRM02 = 1;          /*fclk = fxx /2*/
```

Figure 5-16. Timer 1/Timer 2 Clock Selection Register (PRM02)

	7	6	5	4	3	2	1	0	Address	After reset
PRM02	0	0	0	0	0	0	0	PRM2	FFFFF5D8H	00H

Bit Position	Bit Name	Function
0	PRM2	Specifies the base clock f_{CLK} of timer 1 (TM10) and timer 2 (TM20). 0: $f_{xx} /4$ (when $f_{xx} > 32 \text{ MHz}$ ^{Note}) 1: $f_{xx} /2$ (when $f_{xx} \leq 32 \text{ MHz}$ ^{Note})

★ **Note** This is for the V850E/IA1. It is 40 MHz for the V850E/IA2.

Remark f_{xx} : Internal system clock

(2) Timer 10 noise elimination time selection register (NRC10) settings

In this application circuit example, the NRC10 register is set as follows.

```
NRC10 = 0x03;          /*noise elimination clock selection*/
```

Figure 5-17. Timer 10 Noise Elimination Time Selection Register (NRC10)

	7	6	5	4	3	2	1	0	Address	After reset
NRC10	0	0	0	0	0	0	NRC101	NRC100	FFFFF5F8H	00H

Bit Position	Bit Name	Function															
1, 0	NRC101, NRC100	Selects the TIUD10/TO10, TCUD10/INTP100, and TCLR10/INTP101 pin noise elimination clocks. <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 5px;"> <thead> <tr> <th style="width: 10%;">NRC101</th> <th style="width: 10%;">NRC100</th> <th style="width: 80%;">Noise elimination clocks</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">f_{CLK}/8</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">f_{CLK}/4</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">f_{CLK}/2</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">f_{CLK}</td> </tr> </tbody> </table>	NRC101	NRC100	Noise elimination clocks	0	0	f _{CLK} /8	0	1	f _{CLK} /4	1	0	f _{CLK} /2	1	1	f _{CLK}
NRC101	NRC100	Noise elimination clocks															
0	0	f _{CLK} /8															
0	1	f _{CLK} /4															
1	0	f _{CLK} /2															
1	1	f _{CLK}															
Remark f _{CLK} : Base clock																	

(3) Timer unit mode register 0 (TUM0) settings

In this application circuit example, the TUM0 register is set as follows.

```
TUM0 = 0x80;          /*UDC mode selection*/
```

Figure 5-18. Timer Unit Mode Register 0 (TUM0)

	7	6	5	4	3	2	1	0	Address	After reset
TUM0	CMD	0	0	0	TOE10	ALVT10	0	MSEL	FFFFFF5EBH	00H

Bit Position	Bit Name	Function
7	CMD	Specifies the TM10 operation mode. 0: General-purpose timer mode (up count) 1: UDC mode (up/down count)
3	TOE10	Specifies timer output (TO10) enable. 0: Timer output disabled 1: Timer output enabled Caution When CMD bit = 1 (UDC mode), timer output is not performed regardless of the setting of the TOE10 bit. At this time, timer output consists of the negative phase level of the level set by the ALVT10 bit.
2	ALVT10	Specifies the active level of timer output (TO10). 0: Active level is high level 1: Active level is low level Caution When CMD bit = 1 (UDC mode), timer output is not performed regardless of the setting of the TOE10 bit. At this time, timer output consists of the negative phase level of the level set by the ALVT10 bit.
0	MSEL	Specifies the operation in UDC mode (up/down count) 0: UDC mode A TM10 can be cleared by setting the CLR1, CLR0 bits of the TMC10 register. 1: UDC mode B TM10 is cleared in the following cases. <ul style="list-style-type: none"> • Upon match with CM100 during TM10 up count operation • Upon match with CM101 during TM10 down count operation When UDC mode B is set, the ENMD, CLR1, and CLR0 bits of the TMC10 register become invalid.

(4) Prescaler mode register 10 (PRM10) settings

In this application circuit example, the PRM10 register is set as follows.

```
PRM10 = 0x07;          /*operation mode 4 selection*/
```

Figure 5-19. Prescaler Mode Register 10 (PRM10)

	7	6	5	4	3	2	1	0	Address	After reset
PRM10	0	0	0	0	0	PRM12	PRM11	PRM10	FFFFFF5EEH	07H

Bit Position	Bit Name	Function																																																									
2 to 0	PRM12 to PRM10	<p>Specifies the up/down count operation mode during input of the clock rate when the internal clock of the TM10 is used, or during external clock (TIUD10) input.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th rowspan="2" style="width: 10%;">PRM12</th> <th rowspan="2" style="width: 10%;">PRM11</th> <th rowspan="2" style="width: 10%;">PRM10</th> <th colspan="2" style="width: 40%;">CMD = 0</th> <th colspan="2" style="width: 20%;">CMD = 1</th> </tr> <tr> <th style="width: 10%;">Count Clock</th> <th style="width: 10%;">Count Clock</th> <th style="width: 10%;">Up/Down Count</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td colspan="2" style="text-align: center;">Setting prohibited</td> <td colspan="2" rowspan="4" style="text-align: center;">Setting prohibited (Mode 4) (At this time, the SESA10 register is enabled.)</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td colspan="2" style="text-align: center;">f_{CLK}/2</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td colspan="2" style="text-align: center;">f_{CLK}/4</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td colspan="2" style="text-align: center;">f_{CLK}/8</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td colspan="2" style="text-align: center;">f_{CLK}/16</td> <td rowspan="4" style="text-align: center;">TIUD10</td> <td style="text-align: center;">Mode 1</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td colspan="2" style="text-align: center;">f_{CLK}/32</td> <td style="text-align: center;">Mode 2</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td colspan="2" style="text-align: center;">f_{CLK}/64</td> <td style="text-align: center;">Mode 3</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td colspan="2" style="text-align: center;">f_{CLK}/128</td> <td style="text-align: center;">Mode 4</td> </tr> </tbody> </table> <p>Remark f_{CLK}: Base clock</p>	PRM12	PRM11	PRM10	CMD = 0		CMD = 1		Count Clock	Count Clock	Up/Down Count	0	0	0	Setting prohibited		Setting prohibited (Mode 4) (At this time, the SESA10 register is enabled.)		0	0	1	f _{CLK} /2		0	1	0	f _{CLK} /4		0	1	1	f _{CLK} /8		1	0	0	f _{CLK} /16		TIUD10	Mode 1	1	0	1	f _{CLK} /32		Mode 2	1	1	0	f _{CLK} /64		Mode 3	1	1	1	f _{CLK} /128		Mode 4
PRM12	PRM11	PRM10				CMD = 0		CMD = 1																																																			
			Count Clock	Count Clock	Up/Down Count																																																						
0	0	0	Setting prohibited		Setting prohibited (Mode 4) (At this time, the SESA10 register is enabled.)																																																						
0	0	1	f _{CLK} /2																																																								
0	1	0	f _{CLK} /4																																																								
0	1	1	f _{CLK} /8																																																								
1	0	0	f _{CLK} /16		TIUD10	Mode 1																																																					
1	0	1	f _{CLK} /32			Mode 2																																																					
1	1	0	f _{CLK} /64			Mode 3																																																					
1	1	1	f _{CLK} /128			Mode 4																																																					

(5) Signal edge selection register 10 (SESA10) settings

In this application circuit example, the SESA10 register is set as follows.

```
SESA10 = 0x00;          /*falling edge selection*/
```

Figure 5-20. Signal Edge Selection Register 10 (SESA10) (1/2)

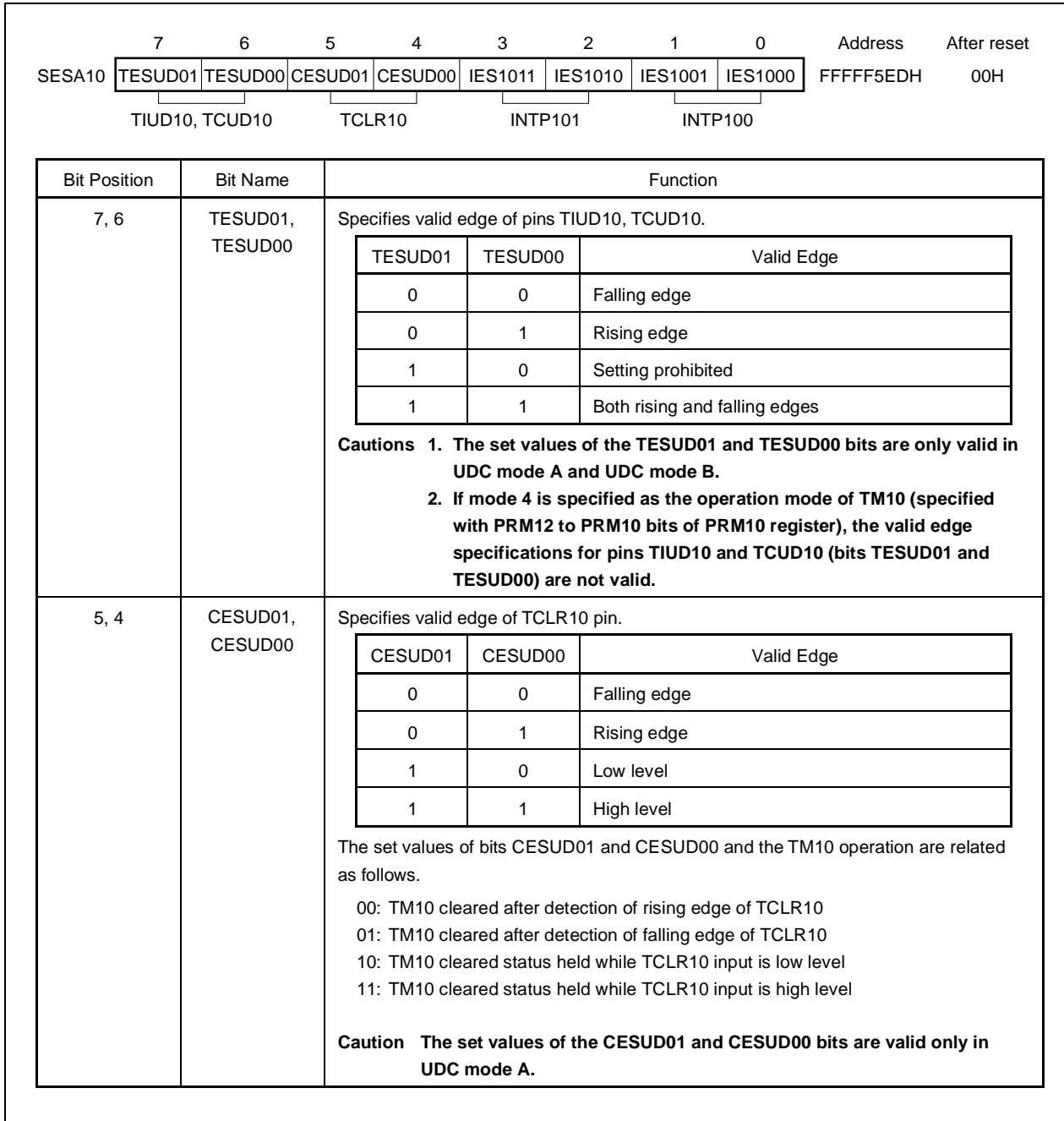


Figure 5-20. Signal Edge Selection Register 10 (SESA10) (2/2)



Bit Position	Bit Name	Function															
3, 2	IES1011, IES1010	<p>Specifies the valid edge of the pin (INTP101/INTP100) that is selected by the CSL0 bit of the CSL10 register.</p> <table border="1"> <thead> <tr> <th>IES1011</th> <th>IES1010</th> <th>Valid Edge</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Falling edge</td> </tr> <tr> <td>0</td> <td>1</td> <td>Rising edge</td> </tr> <tr> <td>1</td> <td>0</td> <td>Setting prohibited</td> </tr> <tr> <td>1</td> <td>1</td> <td>Both rising and falling edges</td> </tr> </tbody> </table>	IES1011	IES1010	Valid Edge	0	0	Falling edge	0	1	Rising edge	1	0	Setting prohibited	1	1	Both rising and falling edges
IES1011	IES1010	Valid Edge															
0	0	Falling edge															
0	1	Rising edge															
1	0	Setting prohibited															
1	1	Both rising and falling edges															
1, 0	IES1001, IES1000	<p>Specifies the valid edge of the INTP100 pin.</p> <table border="1"> <thead> <tr> <th>IES1001</th> <th>IES1000</th> <th>Valid Edge</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Falling edge</td> </tr> <tr> <td>0</td> <td>1</td> <td>Rising edge</td> </tr> <tr> <td>1</td> <td>0</td> <td>Setting prohibited</td> </tr> <tr> <td>1</td> <td>1</td> <td>Both rising and falling edges</td> </tr> </tbody> </table>	IES1001	IES1000	Valid Edge	0	0	Falling edge	0	1	Rising edge	1	0	Setting prohibited	1	1	Both rising and falling edges
IES1001	IES1000	Valid Edge															
0	0	Falling edge															
0	1	Rising edge															
1	0	Setting prohibited															
1	1	Both rising and falling edges															

(6) Timer control register 10 (TMC10) settings

In this application circuit example, the TMC10 register is set as follows.

```
TMC10 = 0x40;          /*count start*/
```

Figure 5-21. Timer Control Register 10 (TMC10)

7	<6>	5	4	3	2	1	0	Address	After reset	
TMC10	0	TM1CE0	0	0	RLEN	ENMD	CLR1	CLR0	FFFFFF5ECH	00H

Bit Position	Bit Name	Function															
6	TM1CE0	Enables/disables TM10 operation. 0: Disable TM10 count operation 1: Enable TM10 count operation															
3	RLEN	Enables/disables transfer from CM100 to TM10. 0: Disable transfer 1: Enable transfer Cautions 1. When RLEN = 1, the value set to CM100 is transferred to TM10 upon occurrence of TM10 underflow. 2. When the CMD bit of the TUM0 register = 0 (general-purpose timer mode), the RLEN bit setting becomes invalid. 3. The RLEN bit is only valid for UDC mode A (TUM0 register CMD bit = 1, MSEL bit = 0). In the general-purpose timer mode (CMD bit = 0) and UDC mode B (CMD bit = 1, MSEL bit = 1), the transfer operation is not performed even though the RLEN bit is set to set (1).															
2	ENMD	Enables/disables clearing of TM10 in general-purpose timer mode (CMD bit of TUM0 register = 0). 0: Clear disable (free running mode) Clearing is not performed even when TM10 and CM100 values match. 1: Clear enable Clearing is performed when TM10 and CM100 values match. Caution When the CMD bit of the TUM0 register = 1 (UDC mode), the ENMD bit setting becomes invalid.															
1, 0	CLR1, CLR0	Controls TM10 clear operation in UDC mode A. <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>CLR1</th> <th>CLR0</th> <th>Specify TM10 Clear Source</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>Clear only by external input (TCLR10)</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td>Clear upon match of TM10 count value and CM100 set value</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>Clear by TCLR10 input or upon match of TM10 count value and CM100 set value</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td>Not cleared</td> </tr> </tbody> </table> Cautions 1. Clearing by match of the TM10 count value and CM100 set value is valid only during TM10 up count operation (TM10 is not cleared during TM10 down count operation). 2. When the CMD bit of the TUM0 register = 0 (general-purpose timer mode), the CLR1 and CLR0 bit settings are invalid. 3. When the MSEL bit of the TUM0 register = 1 (UDC mode B), the CLR1 and CLR0 bit settings are invalid. 4. When clearing by TCLR10 has been enabled with bits CLR1 and CLR0, clearing is performed whether the value of the TM1CE0 bit is 1 or 0.	CLR1	CLR0	Specify TM10 Clear Source	0	0	Clear only by external input (TCLR10)	0	1	Clear upon match of TM10 count value and CM100 set value	1	0	Clear by TCLR10 input or upon match of TM10 count value and CM100 set value	1	1	Not cleared
CLR1	CLR0	Specify TM10 Clear Source															
0	0	Clear only by external input (TCLR10)															
0	1	Clear upon match of TM10 count value and CM100 set value															
1	0	Clear by TCLR10 input or upon match of TM10 count value and CM100 set value															
1	1	Not cleared															

CHAPTER 6 PROGRAM CONFIGURATION

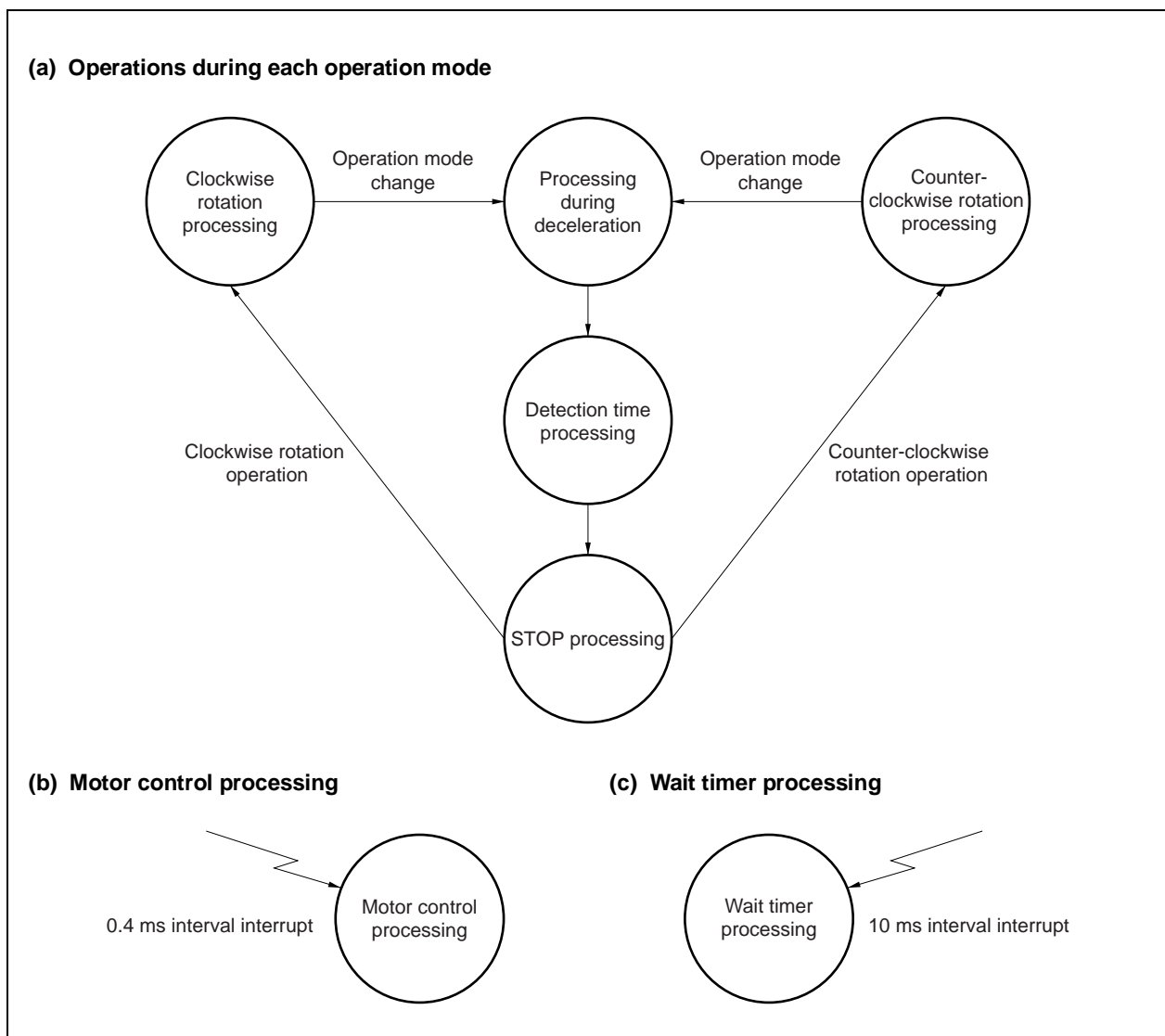
This chapter describes the program configuration used for the application circuit example.

- ★ This chapter, **CHAPTER 7 FLOW CHARTS**, and **CHAPTER 8 PROGRAM LISTS** describe application circuit examples using the V850E/IA1. When using the V850E/IA2, perform design making allowances for differences in functions between the V850E/IA1 and V850E/IA2 such as port timer, and timing settings.

6.1 Program Structure

The program structure is shown in Figure 6-1.

Figure 6-1. Program Structure



[Description of structure]

- (1) During the main processing, the operation mode switch status is monitored and the status changes between clockwise rotation operation, counter-clockwise rotation operation, and stop operation modes are controlled.
- (2) As the motor is controlled, its status is indicated via 0.4 ms interval interrupts.
- (3) The motor can be controlled in the following three ways.
 - Origin detection rotation
Detects origin using constant torque and constant rotation speed.
 - Speed control
Rotation speed (rpm) is maintained at the specified speed.
 - Position control
The specified position is maintained.
- (4) Read A/D values
A/D conversion of the U-phase current value (ANI00) and V-phase current value (ANI10) begins at the start of 0.4 ms interval interrupt servicing and the converted values are input and used in current feedback processing. A/D conversion of the volume value input (ANI01) begins after the current value is input, and input of converted values occurs after conversion processing. The values are set to the volume area and are used in main processing.

6.2 Common Areas

The common areas used in the application circuit example are described in Table 6-1.

Table 6-1. Common Area List

Symbol	Type	Application	Setting Value
stop_req	unsigned char	Request to monitor motor rotation stop: see (3)	ON: Request sent OFF: No request
stop_flag	unsigned char	Stop motor rotation: see (4)	ON: Stopped OFF: Rotating
init_flag	unsigned char	Rotating to detect motor's origin: see (1)	ON: Rotating to detect motor's origin OFF: Normal rotation
cont_mode	unsigned char	Motor's control method: see (2)	POSITION: Position control SPEED: Speed control
error_flag	unsigned char	Error	0: No error ERR_NO1: Overcurrent ERR_NO2: Position error ERR_NO3: Drive error
timer_count	unsigned short	Wait time measurement counter	Set value, timeout occurs at zero Unit: 10 ms
volume	unsigned short	Stores A/D value of speed setting volume: see (5)	0 to 1,023
before_enc	signed short	Stores previous encoder value: see (6)	-9,999 to 9,999
now_position	signed int	Current motor position	Unit: pulse
o_position	signed int	Target motor position: see (7)	Unit: pulse
now_speed	signed int	Current speed	Unit: pulse/0.4 ms
o_speed	signed int	Target speed: see (8)	Unit: pulse/0.4 ms
i_speed	signed int	Area for storing speed integral	-
o_trm	signed int	Target position for origin detection rotation: see (9)	Unit: pulse

[Description of main common areas]

(1) init_flag

The motor is rotated twice at a constant torque to detect the Z phase so that the absolute positions of the motor's axes can be determined.

init_flag is set to ON during this rotation to detect the origin, and it is used for origin detection rotations as motor control processing.

(2) cont_mode

In the application circuit example's motor control processing, either the speed or the position is controlled. cont_mode is used to select the control method.

(3) stop_req

If the operation mode is changed during operation in either the clockwise or counter-clockwise direction, operation is temporarily stopped before starting again in the newly set operation mode.

(4) stop_flag

This flag is used to report the end of origin detection rotation and motor stoppage during normal rotation that occurs after a monitoring stop request has been issued.

(5) volume

The speed setting volume's A/D value is stored and the following equation is used to calculate the speed setting.

$$\text{Speed (pulse/0.4 ms)} = \{ (\text{SPEED_MAX} \times \text{volume}) / 1,024 \} + 1$$

(6) before_enc

The previous encoder value is stored in this area and the following equation is used to calculate the differential (speed variation).

$$\text{sa_enc (differential)} = \text{current encoder value} - \text{previous encoder value}$$

When the absolute value of "sa_enc" is greater than "1 rotation encoder/2," it is regarded as beyond the zero point, in which case sa_enc is compensated accordingly.

- sa_enc < 0
sa_enc = sa_enc + Encoders per rotation
- sa_enc ≥ 0
sa_enc = sa_enc – Encoders per rotation

(7) o_position

When in position control mode (STOP mode), the control function uses this position as the return position.

(8) o_speed

When in speed control mode (during clockwise or counter-clockwise operation), the speed is controlled at this setting. The speed setting volume calculates and sets this speed setting.

(9) o_trm

This area is used to set the target transfer position for origin detection rotation. This value sets the amount of rotation needed to reach the second full rotation (20,000 pulses).

6.3 Constant Definition

Constants used in the application circuit example are listed below.

Table 6-2. List of Constants

Symbol	Application	Constant
ON	Flag ON	1
OFF	Flag OFF	0
CW	Clockwise rotation mode	1
CCW	Counter-clockwise rotation mode	2
STOP	Stop mode	0
SPEED	Speed control mode	0
POSITION	Position control mode	1
ERR_NO1	Overcurrent error	1
ERR_NO2	Position error	3
ERR_NO3	Drive error	7
P	Motor poles: see (1)	4
MAXPULSE	Encoder count per motor rotation: see (2)	10,000 (pulses)
OFFSET	Offset value for motor origin: see (3)	800 (pulses)
SPEED_MAX	Motor's maximum rotation speed: see (4)	100 (pulses/0.4 ms)
VMAX	Maximum voltage applied to motor: see (5)	100 (V)
IS_MAX	Maximum speed integral value: see (6)	2,000,000
MAX_I	Maximum current value: see (7)	400 (mA)
SA_POSI_MAX	Maximum position displacement: see (8)	5,000 (pulses)
★ CM3_DATA	Timer 00 (TM00) CM003 setting value: see (9)	625
★ BFCM_DATA	Timer 00 (TM00) BFCM maximum setting value: see (10)	625
KPGETA	Position gain offset: see (11)	10
KSIGETA	Speed integral gain offset: see (11)	8
KIGETA	Voltage conversion gain offset: see (11)	12
SGETA	sin value offset: see (11)	14

[Description of main constants]

(1) P

This is the number of poles per motor rotation, as used in this application circuit example. One rotation of the motor equals the waveform for four electrical rotations.

(2) MAXPULSE

This indicates the number of encoder pulses per motor rotation.
 1 rotation (2,500 pulses) × 4 = 10,000 pulses

(3) OFFSET

This indicates the offset between the motor's Z phase and the coil position.

(4) SPEED_MAX

This is the maximum rotation speed of motor, as used in this application circuit example.

$$(100 \times 60 / 0.4 \text{ ms}) / 10,000 \text{ pulses} = 1,500 \text{ rpm}$$

(5) VMAX

This is the amount of voltage that is applied when PWM reaches 100%. The applied voltage corresponding to 100% PWM remains the same even if the calculated VMAX value is greater than that.

(6) IS_MAX

This is the maximum speed integral value. This value is the highest value possible for IS_MAX.

(7) MAX_I

This is the maximum current value. An overcurrent occurs if the MAX_I value is exceeded.

(8) SA_POSI_MAX

This is the maximum position displacement value. If the position displacement exceeds the SA_POSI_MAX value, a position error occurs.

(9) CM3_DATA

This is the value of timer 00 (TM00)'s compare register 003 (CM003).

The CM3_DATA value determines the 20 kHz carrier frequency.

(10) BFCM_DATA

This is the maximum value when "100 V" has been specified for timer 00 (TM00)'s buffer registers CM00 to CM02 (BFCM00 to BFCM02)

(11) KPGETA, KSIGETA, KIGETA, SGETA

These are offsets for various operations.

For example, in 2ⁿ the n indicates the offset.

6.4 Motor Control Constants

The motor control constants that are used in this application circuit example are listed below.

Table 6-3. List of Motor Control Constants

Symbol	Type	Application	Constant Value
kp	signed int	Position-proportional gain	500
ksp	signed int	Speed-proportional gain	100
ksi	signed int	Speed-integral gain	100
ki	signed int	Voltage conversion gain	11

[Description of constants]

(1) kp

This is the proportional gain used to convert from the position differential to speed.
The offset value is 2^{10} .

(2) ksp

This is the proportional gain used to convert from speed to current.

(3) ksi

This is the speed's integral gain.
The offset value is 2^8 .

(4) ki

This is the gain used to convert from current to voltage.
The offset value is 2^{12} .

CHAPTER 7 FLOW CHARTS

This chapter presents flow charts of various processing in the application circuit example using the V850E/IA1.

7.1 Main Routine

Figure 7-1 illustrates the flow of the main routine.

Figure 7-1. Main Routine (1/6)

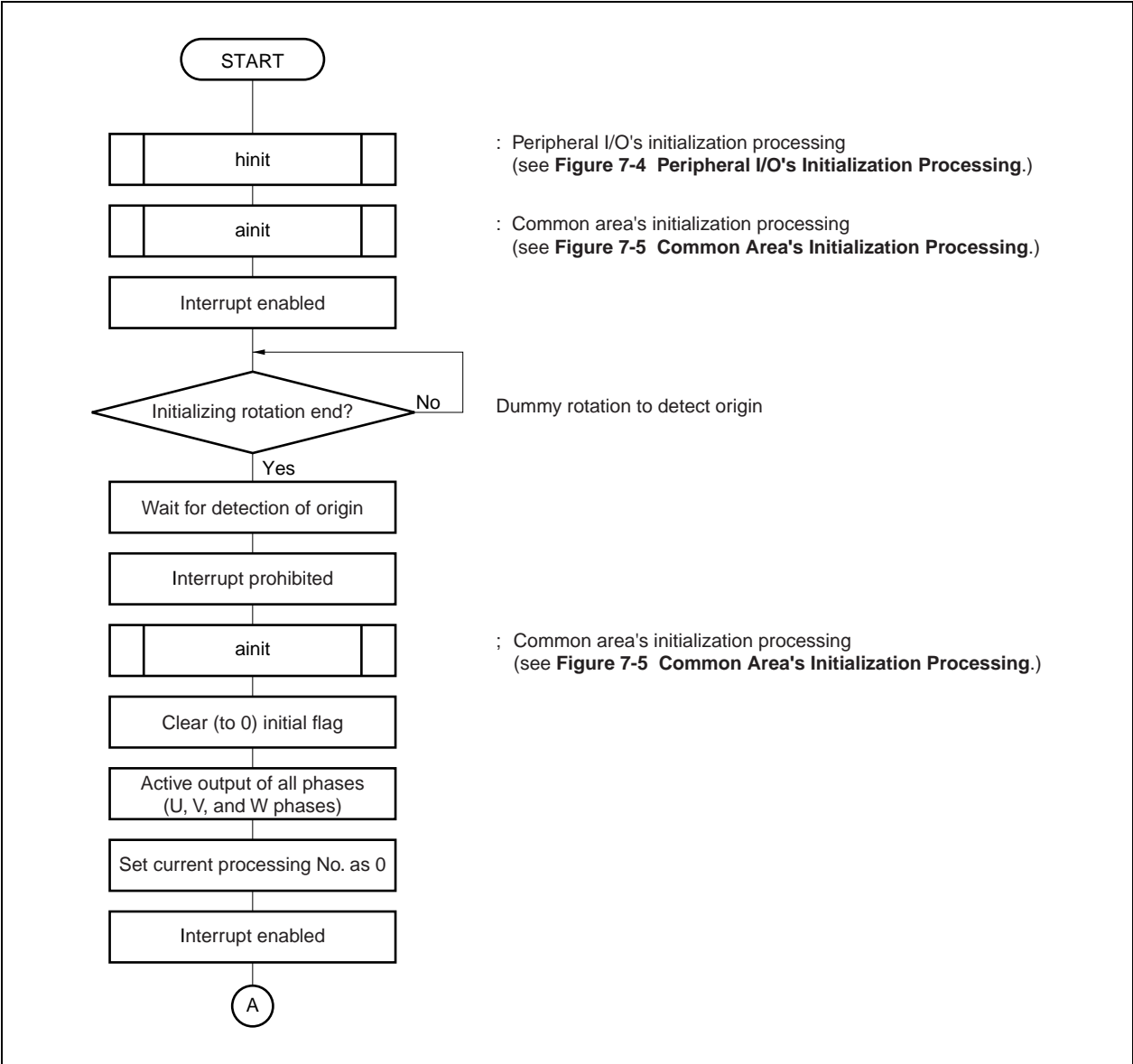


Figure 7-1. Main Routine (2/6)

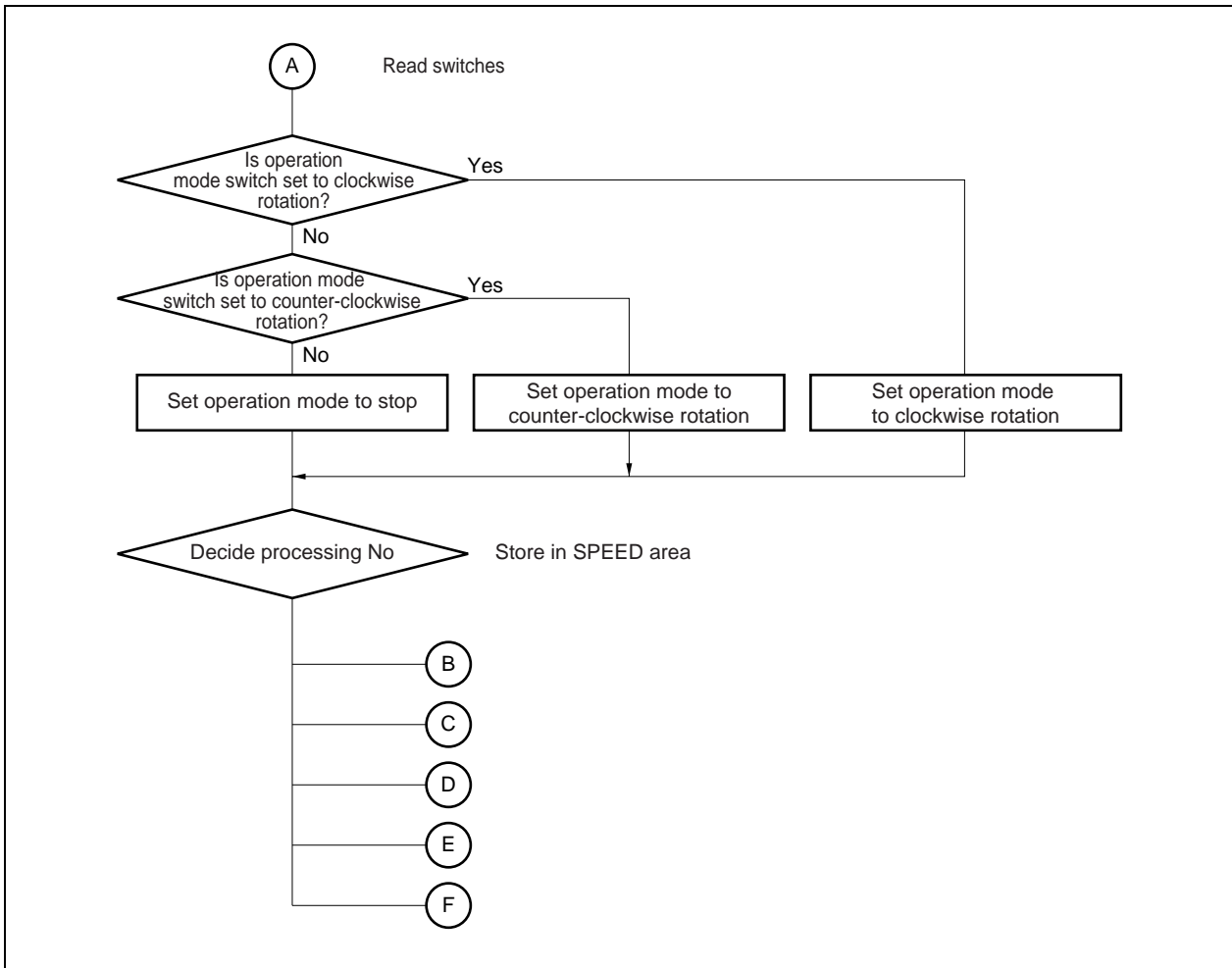


Figure 7-1. Main Routine (3/6)

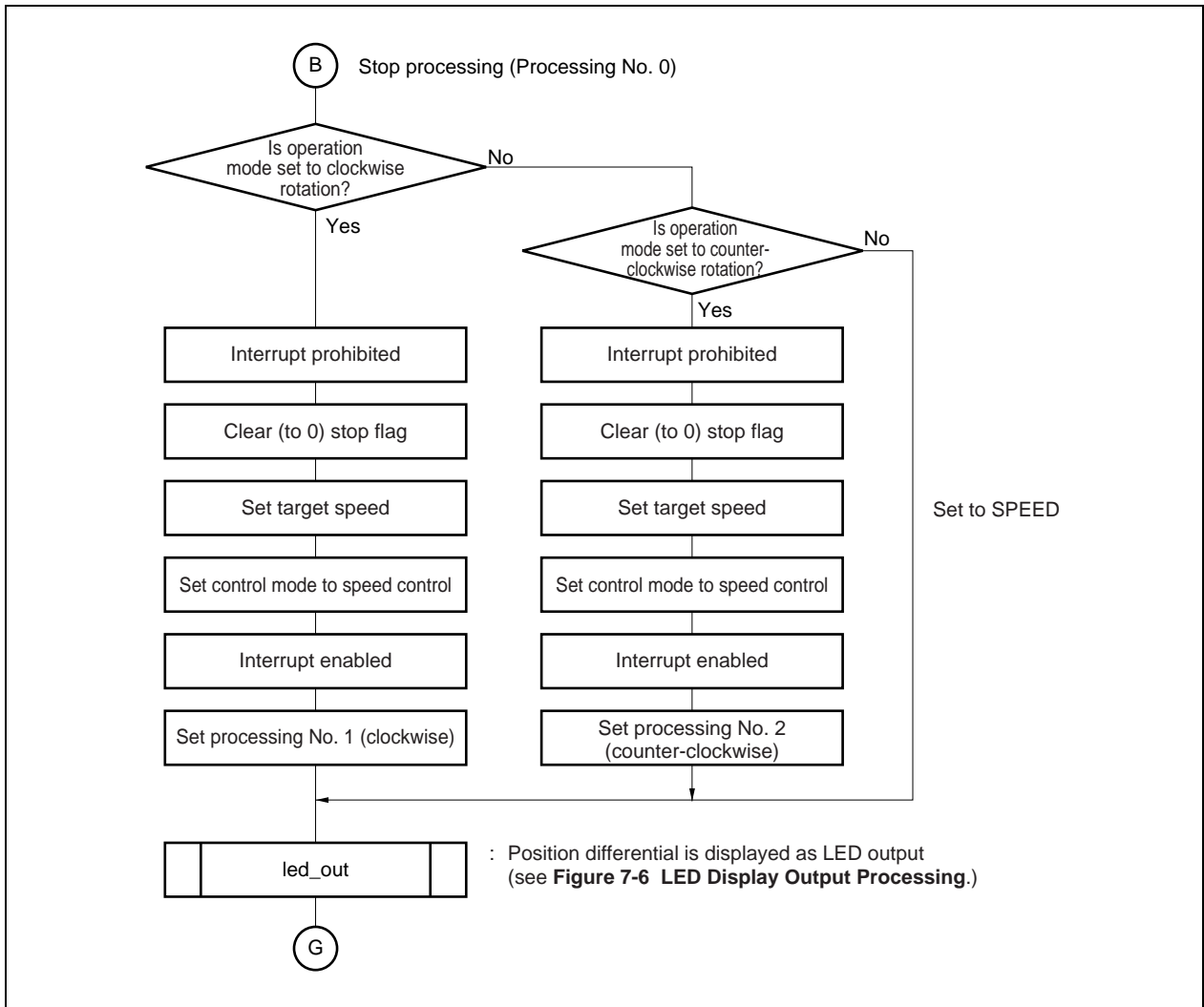


Figure 7-1. Main Routine (4/6)

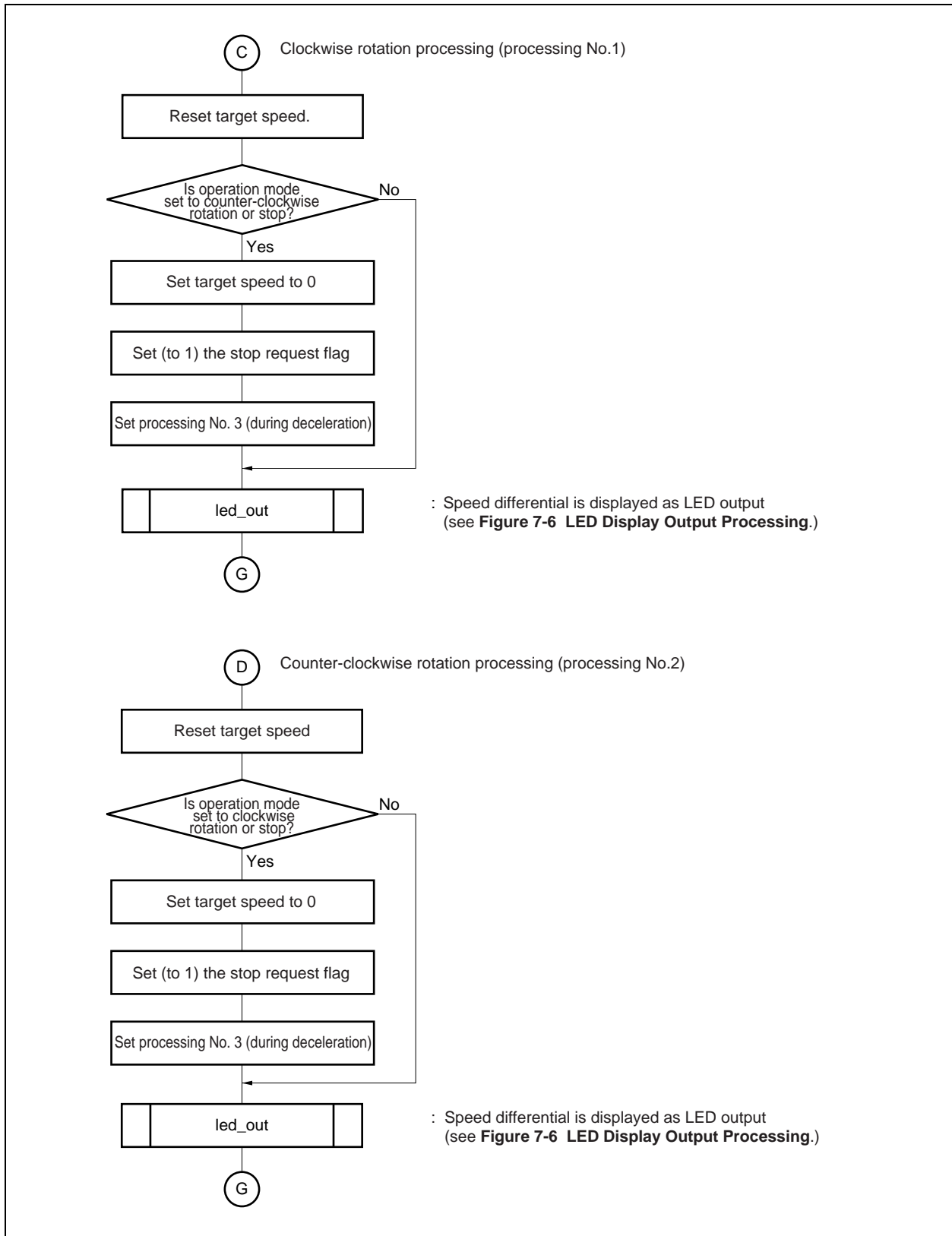


Figure 7-1. Main Routine (5/6)

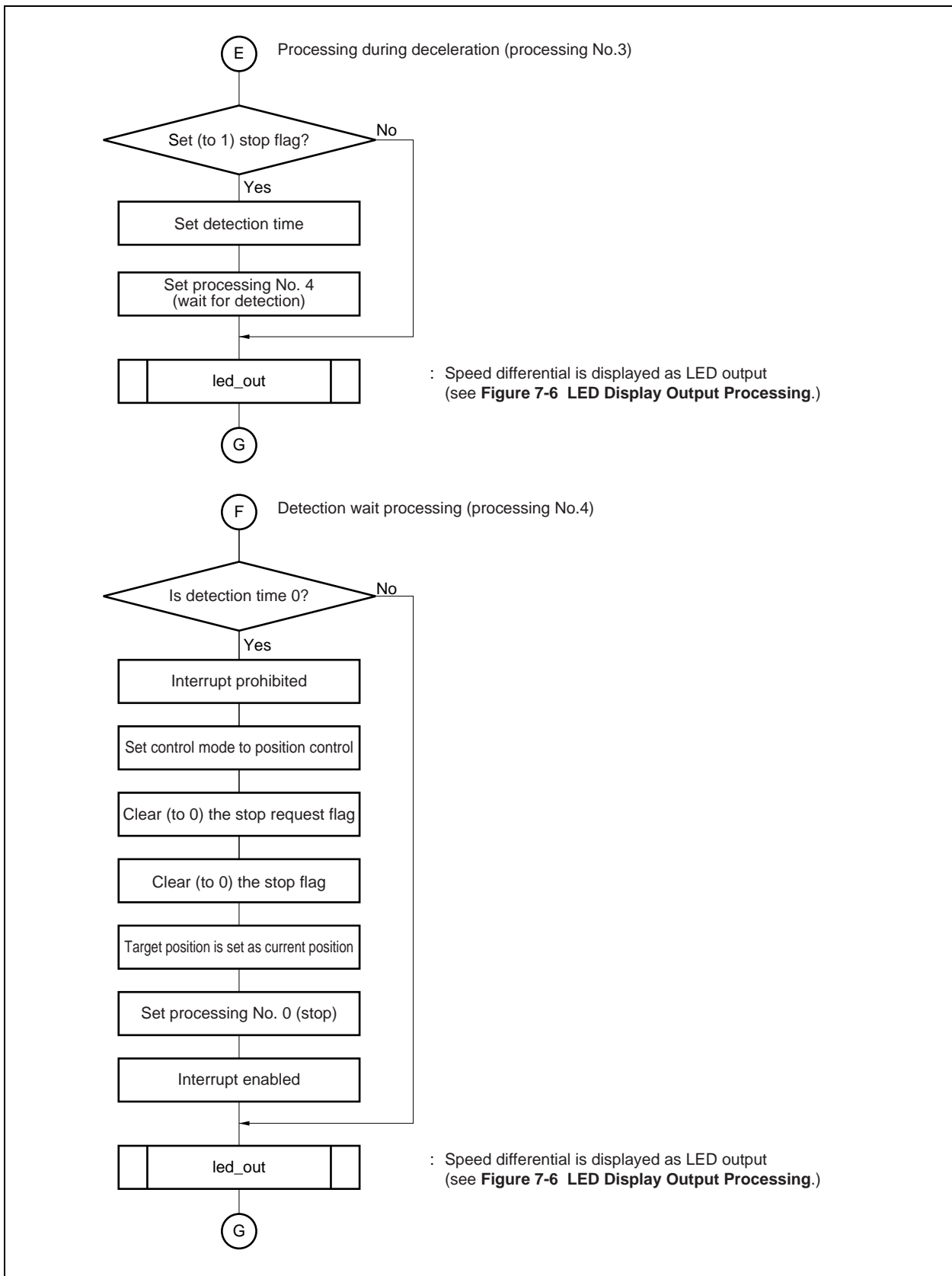
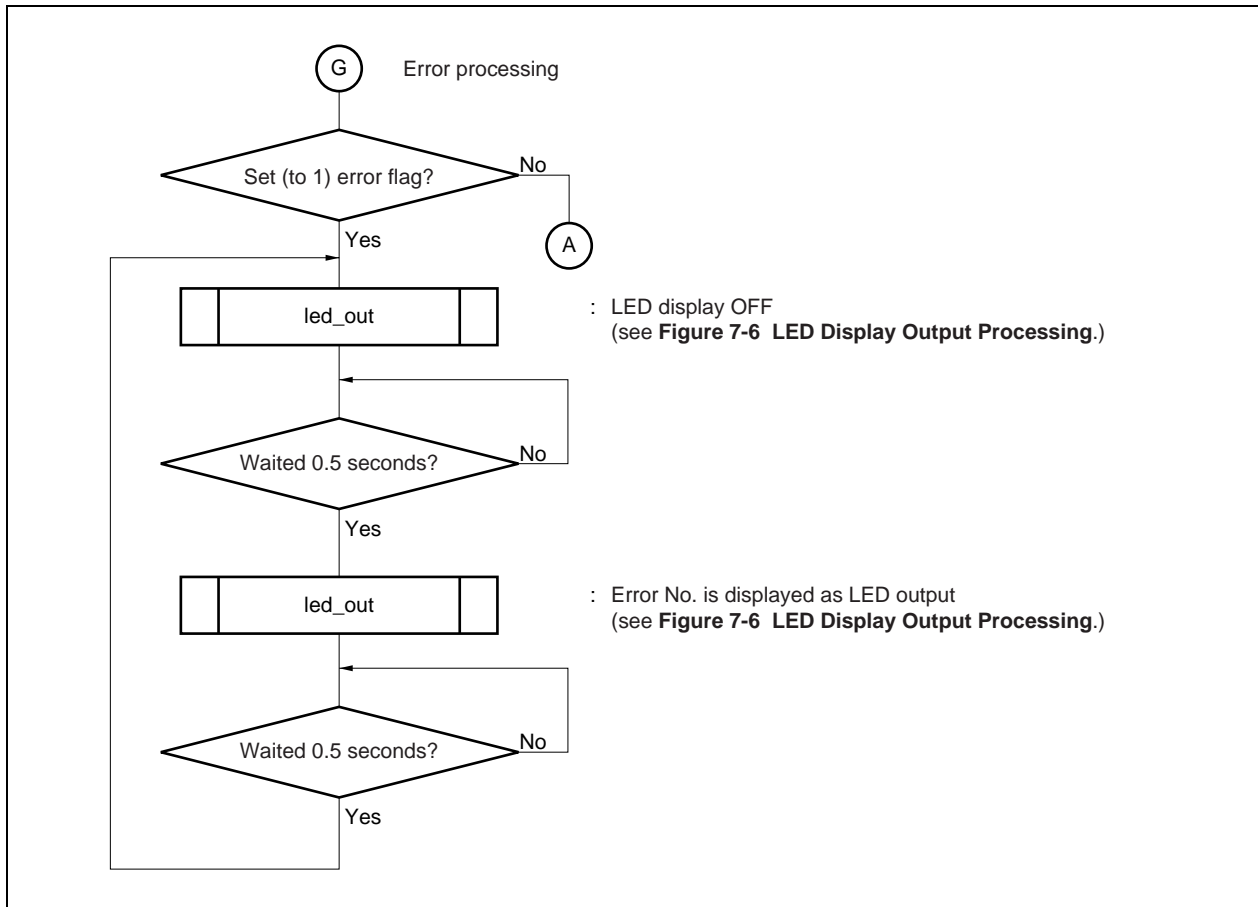


Figure 7-1. Main Routine (6/6)



7.2 Motor Control Interrupt Servicing (0.4 ms Interval)

Figure 7-2 illustrates the flow of motor control interrupt servicing (0.4 ms interval).

Figure 7-2. Motor Control Interrupt Servicing (0.4 ms Interval) (1/5)

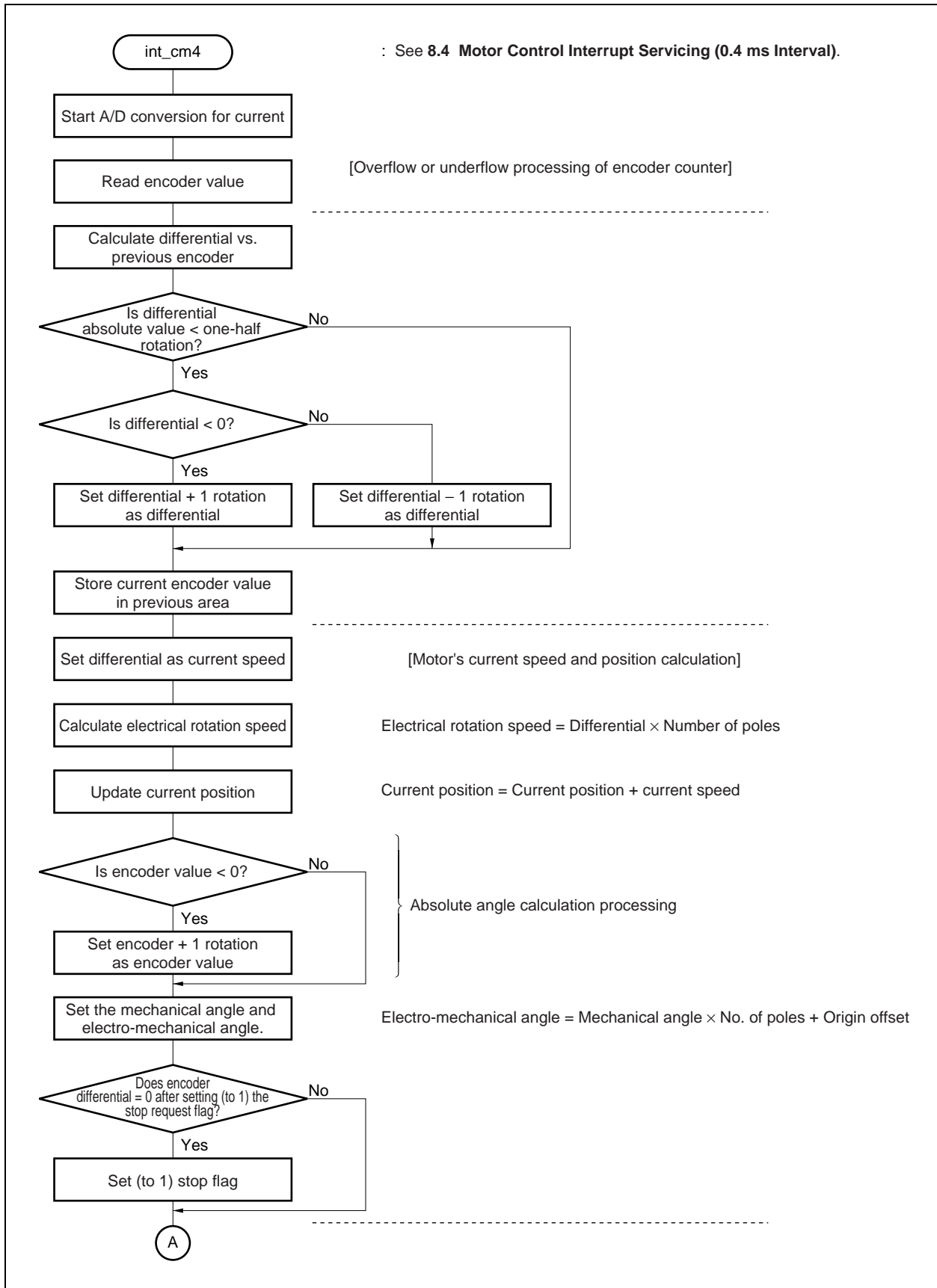


Figure 7-2. Motor Control Interrupt Servicing (0.4 ms Interval) (2/5)

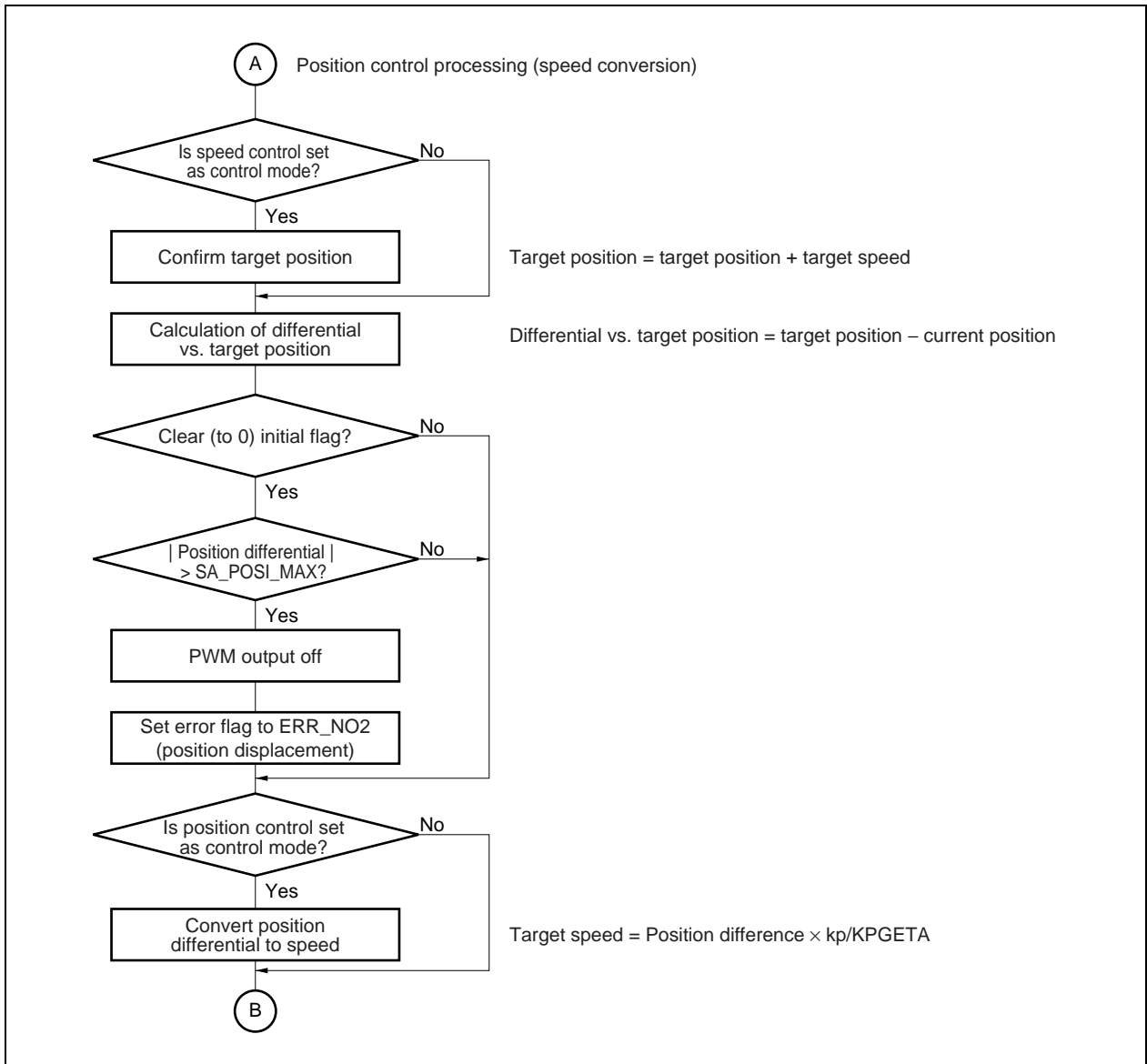


Figure 7-2. Motor Control Interrupt Servicing (0.4 ms Interval) (3/5)

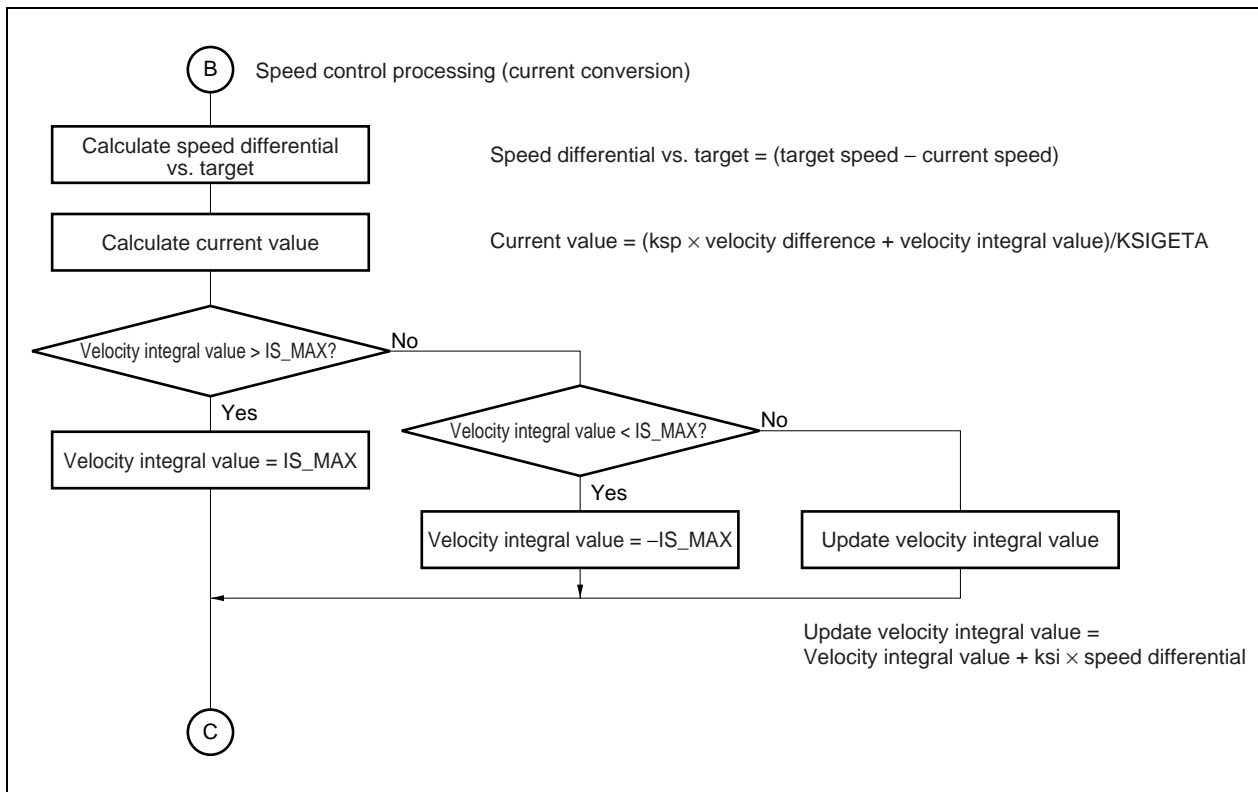


Figure 7-2. Motor Control Interrupt Servicing (0.4 ms Interval) (4/5)

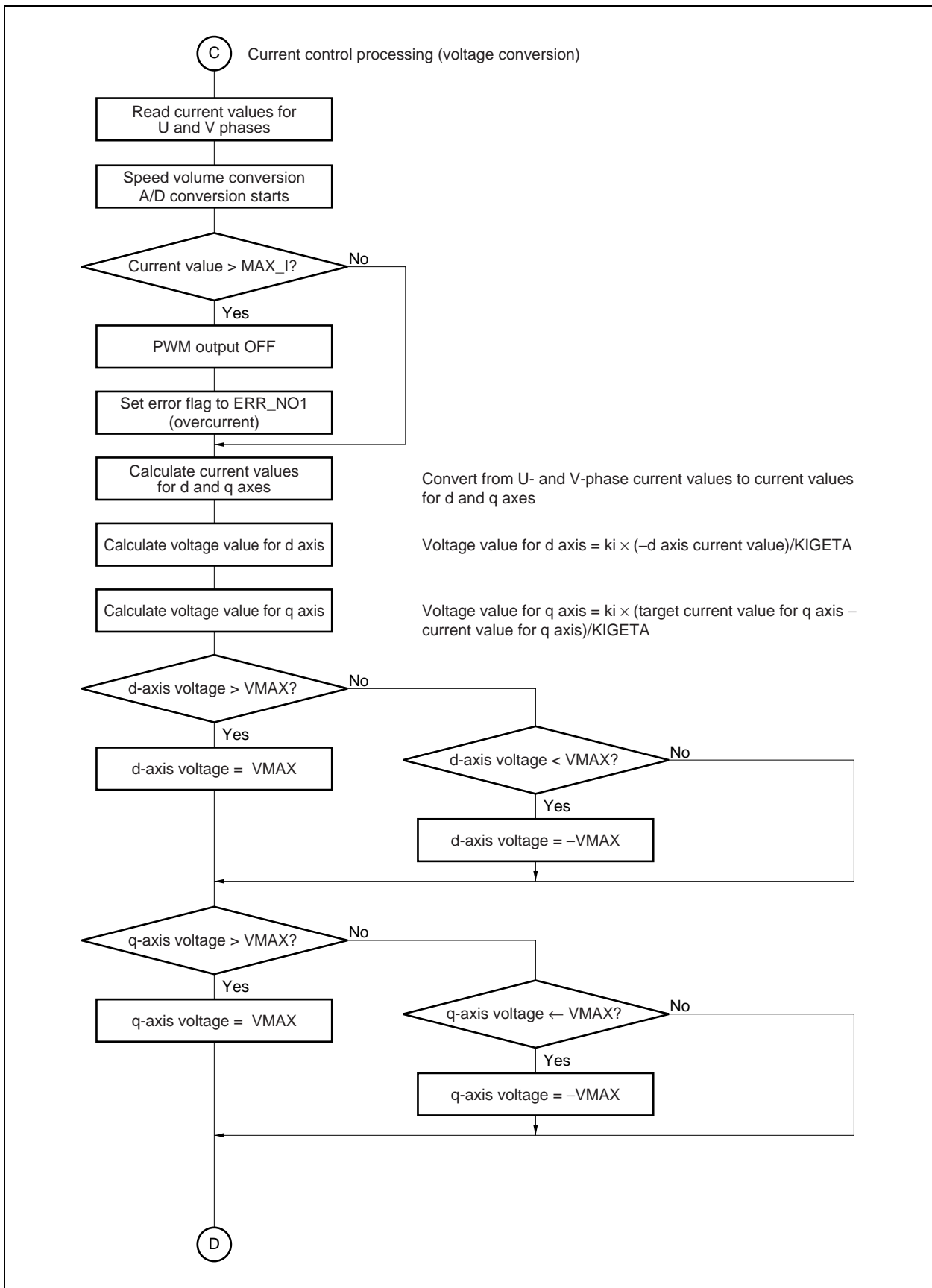
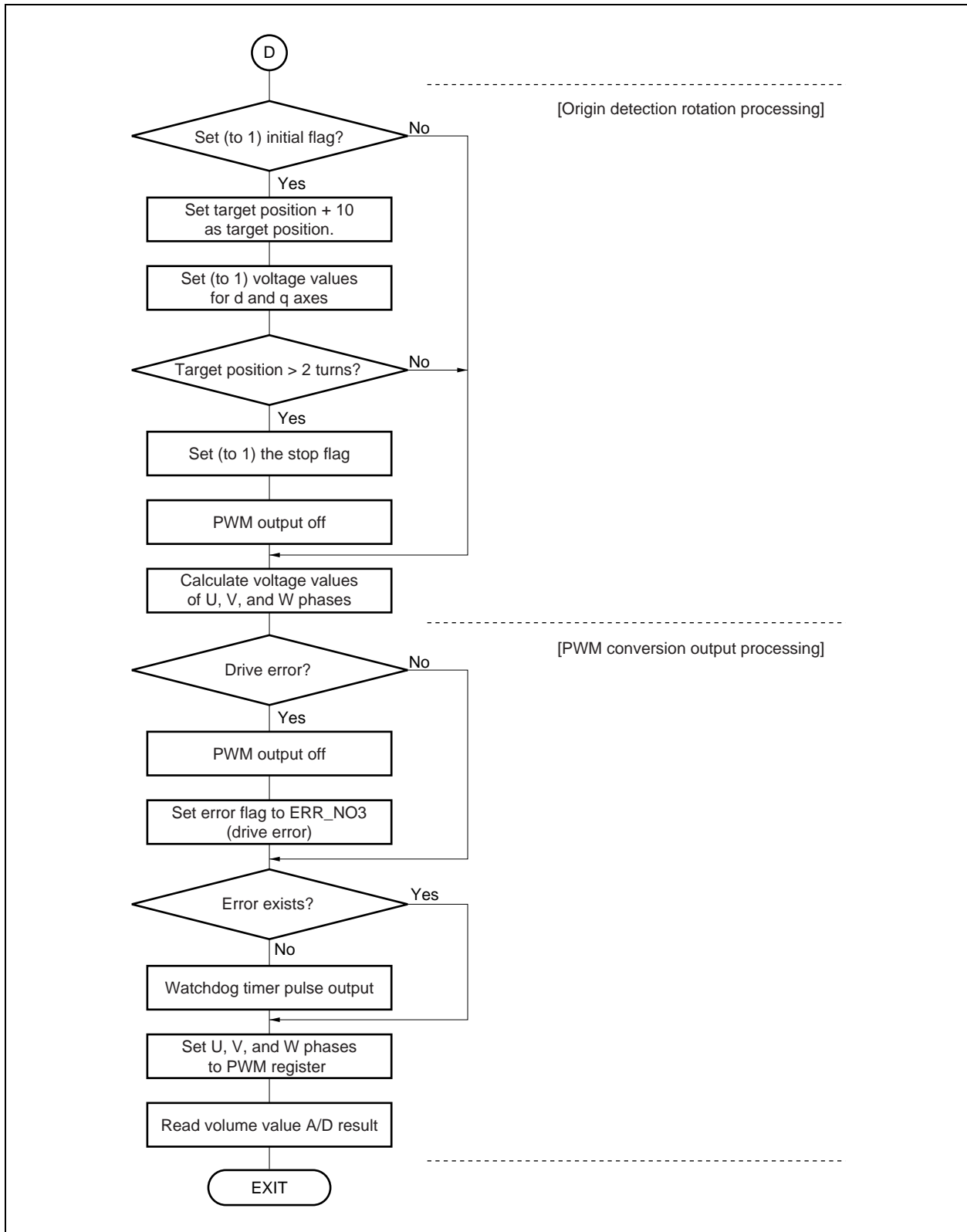


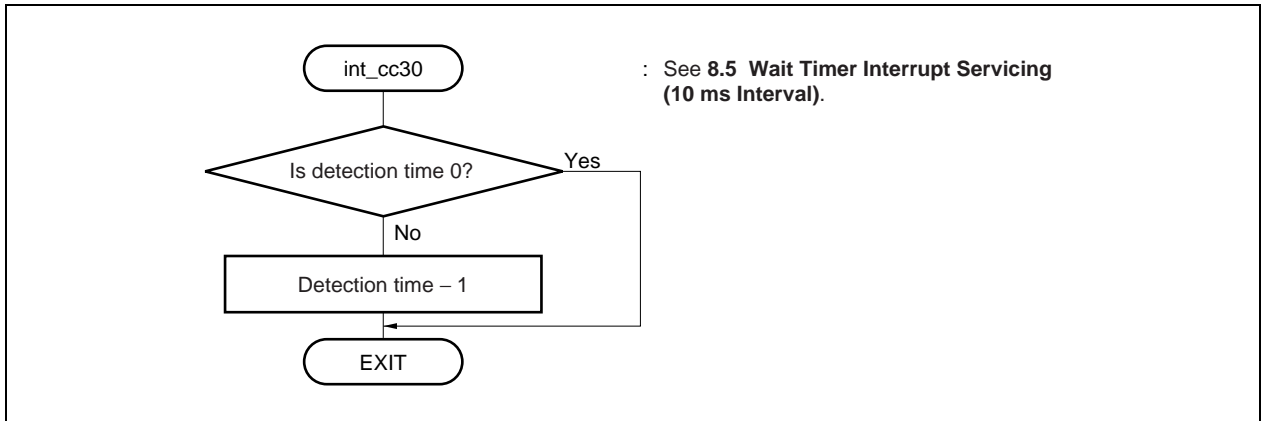
Figure 7-2. Motor Control Interrupt Servicing (0.4 ms Interval) (5/5)



7.3 Wait Timer Interrupt Servicing (10 ms Interval)

Figure 7-3 illustrates the flow of wait timer interrupts (10 ms interval).

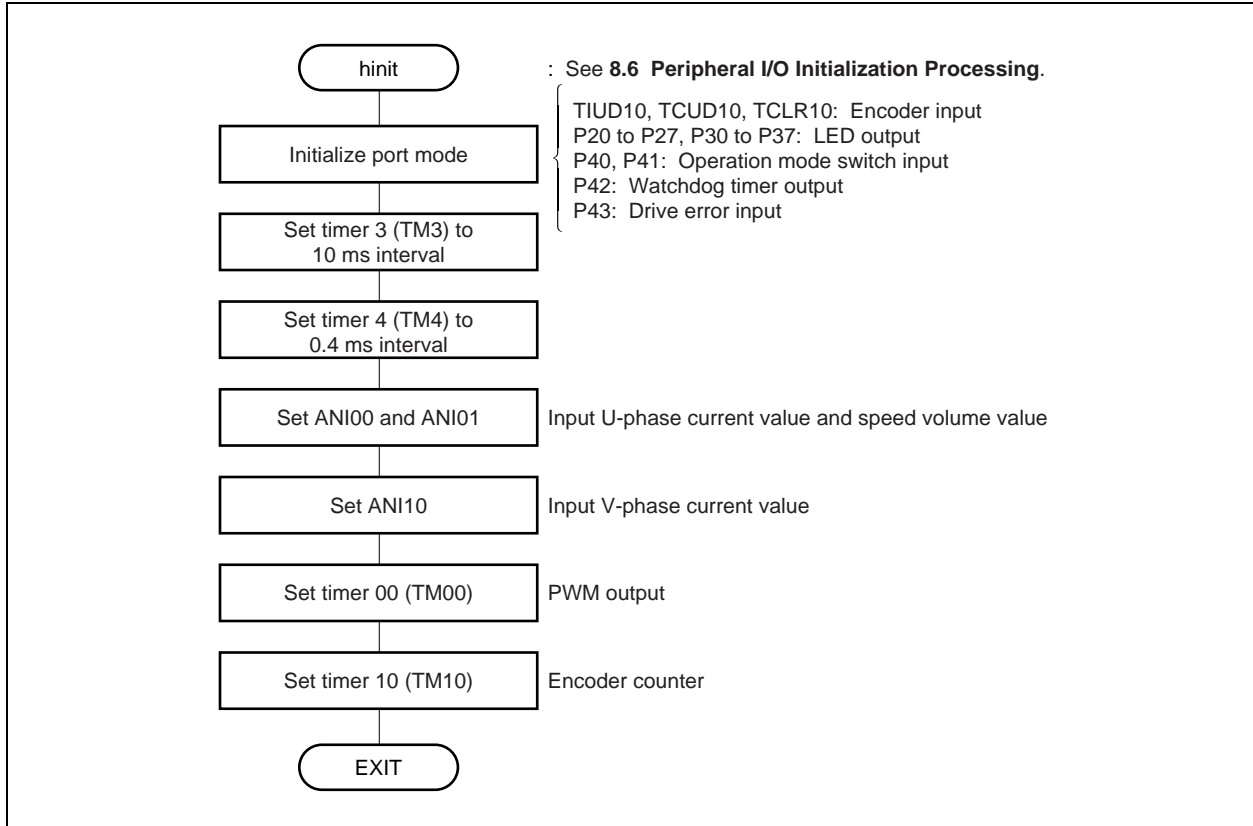
Figure 7-3. Wait Timer Interrupt Servicing (10 ms Interval)



7.4 Peripheral I/O Initialization Processing

Figure 7-4 illustrates the flow of peripheral I/O initialization processing.

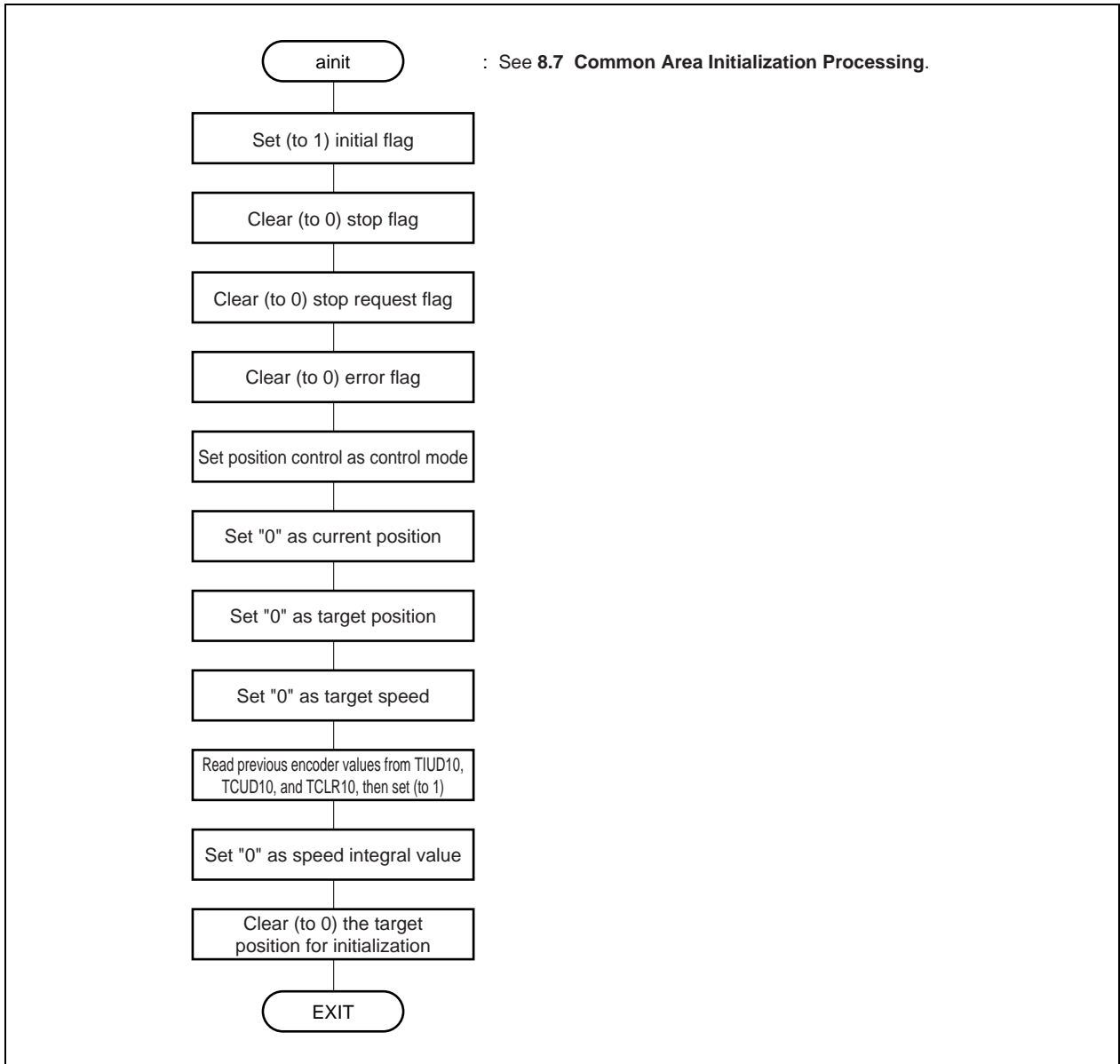
Figure 7-4. Peripheral I/O Initialization Processing



7.5 Common Area Initialization Processing

Figure 7-5 illustrates the flow of common area initialization processing.

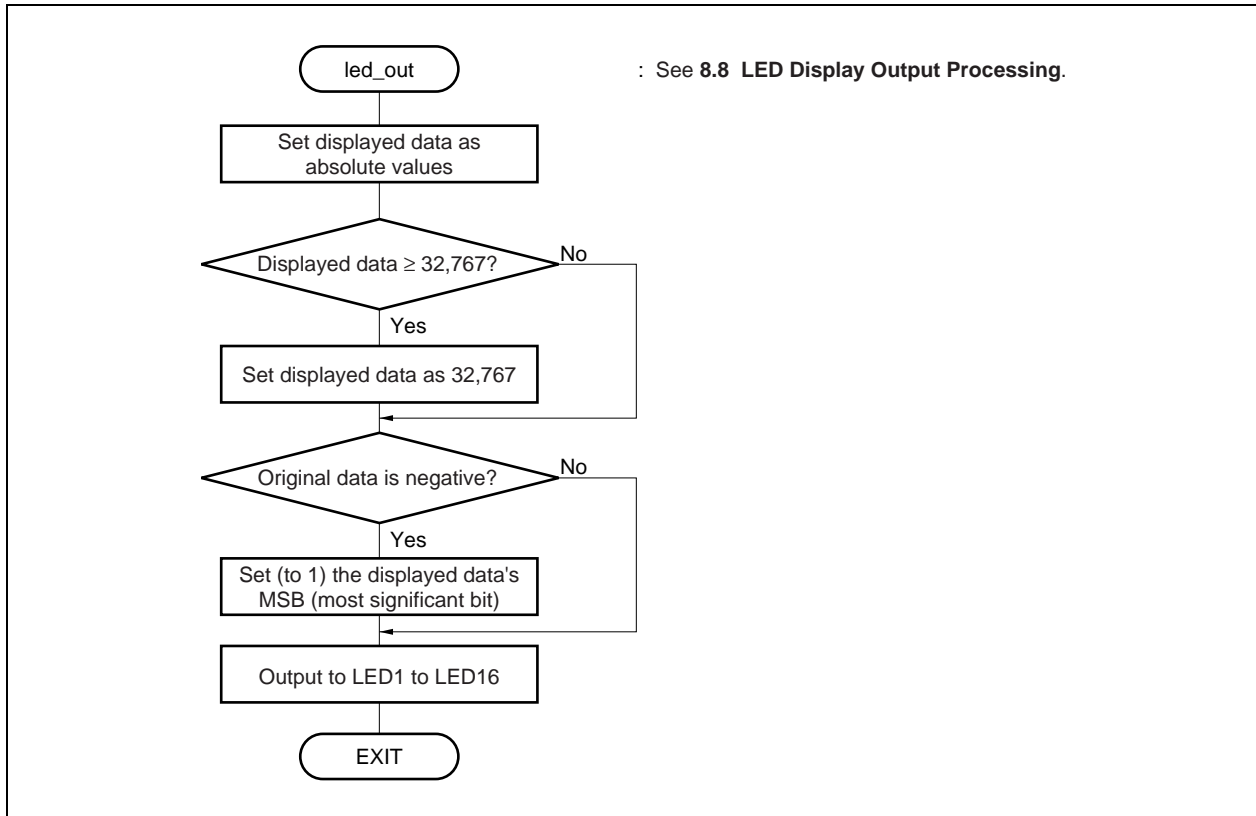
Figure 7-5. Common Area Initialization Processing



7.6 LED Display Output Processing

Figure 7-6 illustrates the flow of LED display output processing.

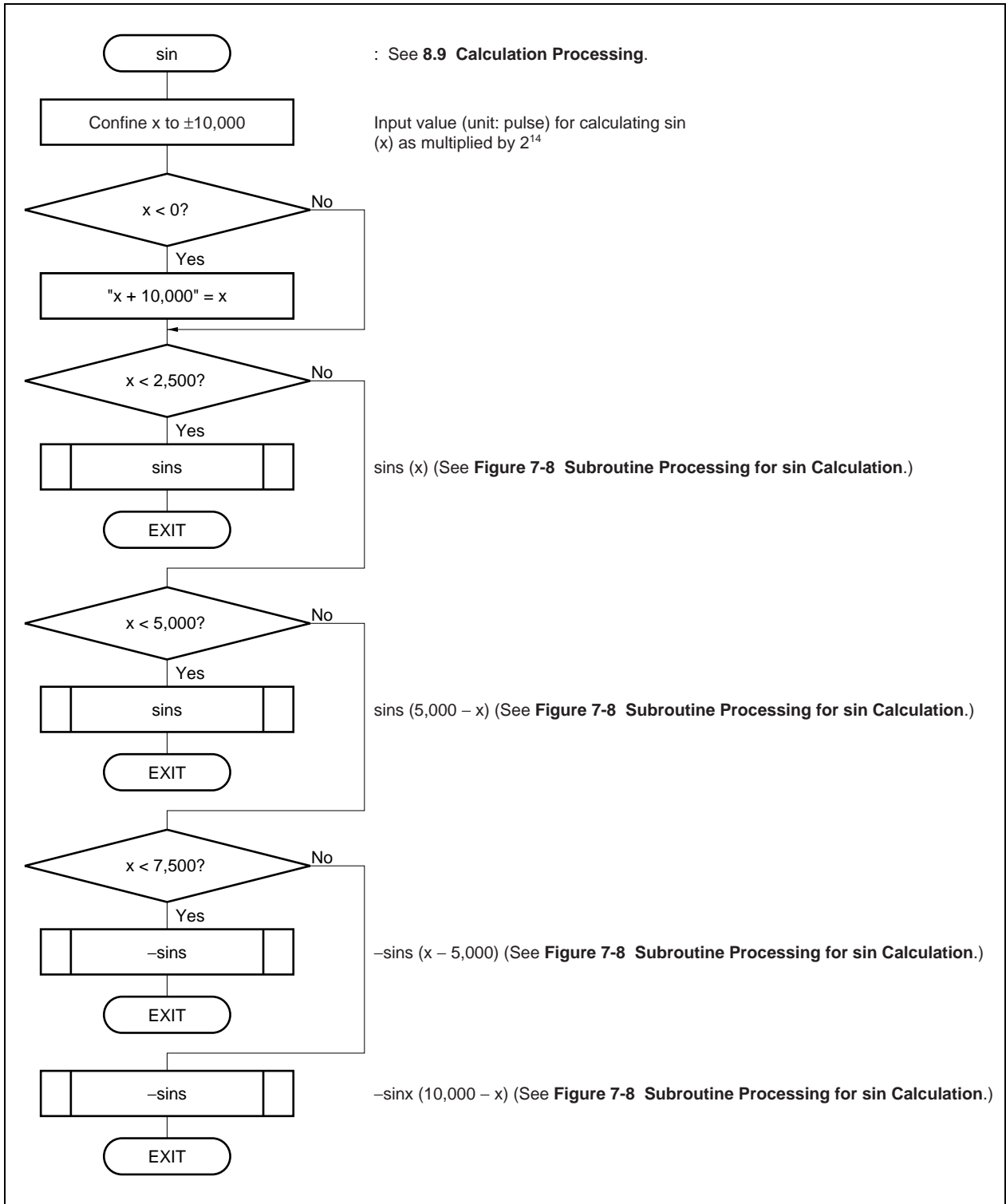
Figure 7-6. LED Display Output Processing



7.7 sin Calculation Processing

Figure 7-7 illustrates the flow of sin calculation processing.

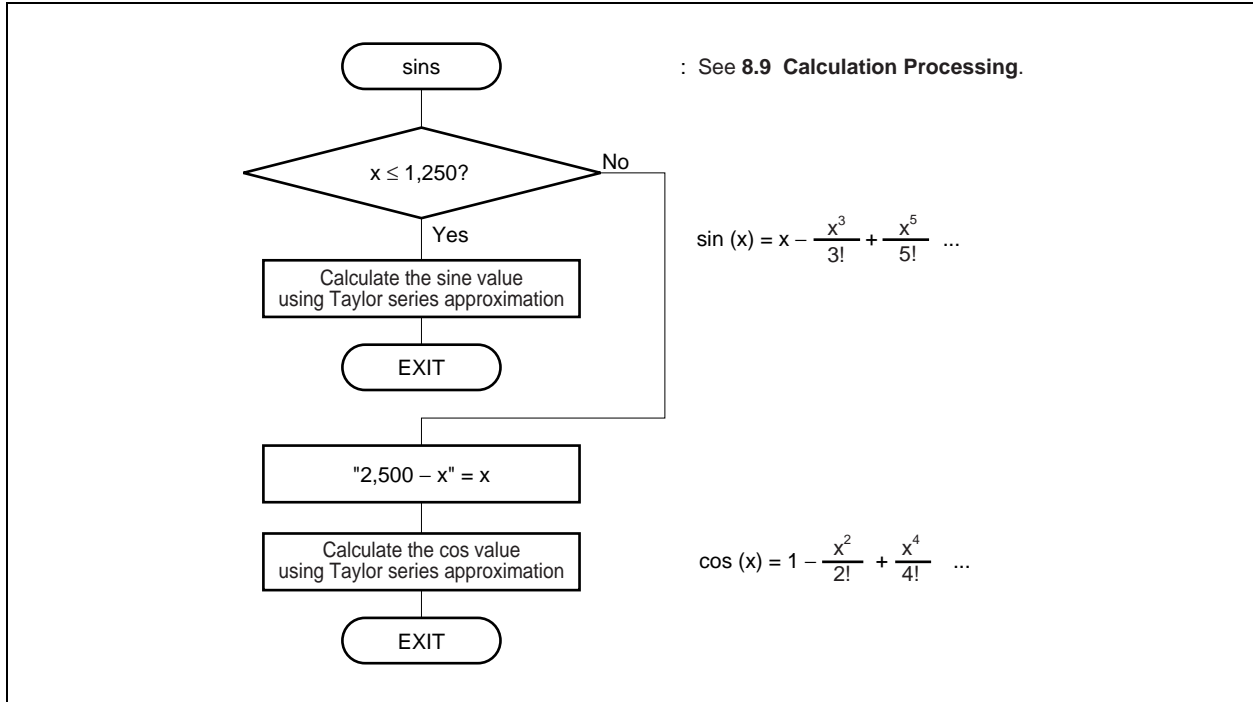
Figure 7-7. sin Calculation Processing



7.8 Subroutine Processing for sin Calculation

Figure 7-8 illustrates the subroutine processing for the sin calculation.

Figure 7-8. Subroutine Processing for sin Calculation



CHAPTER 8 PROGRAM LISTS

This chapter presents program lists from the application circuit example using the V850E/IA1.

8.1 Definition of Constants

```
#pragma ioreg                                /* Peripheral I/O register definition */
#pragma interrupt INTCC30 int_cc30           /* Register to interrupt handler address */
#pragma interrupt INTCM4 int_cm4            /* Register to interrupt handler address */
/*****
/* Constant definitions
*****/
#define ON 1
#define OFF 0
#define CW 1 /* Clockwise rotation operation mode */
#define CCW 2 /* Counter-clockwise rotation operation mode */
#define STOP 0 /* Stop mode */
#define SPEED 0 /* Speed control mode */
#define POSITION 1 /* Position control mode */
#define ERR_NO1 1 /* Overcurrent error */
#define ERR_NO2 3 /* Position error */
#define ERR_NO3 7 /* Drive error */
/**** Motor constants *****/
#define P 4 /* No. of poles */
#define MAXPULSE 10000 /* Pulses per rotation */
#define OFFSET 800 /* Origin offset */
#define SPEED_MAX 100 /* Maximum speed pulse/0.4 ms */
#define VMAX 100 /* Maximum voltage 100 V */
#define IS_MAX 2000000 /* Maximum speed integral value */
#define MAX_I 400 /* Current maximum value */
#define SA_POSI_MAX 5000 /* Maximum position differential pulse */
#define CM3_DATA 625 /* PWM CM003 setting value 20 kHz */
#define BFCM_DATA 625 /* PWM PFCM setting maximum value */
#define KPGETA 10 /* kp offset */
#define KSIGETA 8 /* ksi offset */
#define KIGETA 12 /* ki offset */
#define SGETA 14 /* sin offset */
/**** Motor constants *****/
#pragma section const begin
signed int kp = 500 ; /* Position-proportional gain */
signed int ksp = 100 ; /* Speed-proportional gain */
signed int ksi = 100 ; /* Speed integral gain */
signed int ki = 11 ; /* Current conversion constant */
#pragma section const end
```

8.2 Common Area

```
/******  
/*  
/* Common area  
/*  
/******  
unsigned char  init_flag ;           /* Initial flag */  
unsigned char  cont_mode ;          /* Control mode  0: Speed mode, 1: Position mode */  
unsigned char  stop_req ;           /* Stop request flag */  
unsigned char  stop_flag ;         /* Stop flag */  
unsigned char  error_flag ;        /* Error flag */  
unsigned short timer_count ;       /* Counter for measurement of detection time */  
unsigned short volume ;           /* Speed setting volume value */  
signed short  before_enc ;        /* Previous encoder value */  
signed   int   now_position ;      /* Current position (pulses) */  
signed   int   o_position ;       /* Target position (pulses) */  
signed   int   now_speed ;        /* Current speed (pulses/ms) */  
signed   int   o_speed ;         /* Target speed (pulses/ms) */  
signed   int   i_speed ;         /* Speed integral value area */  
signed   int   o_trm ;           /* Target position for initialization */
```

8.3 Main Routine

```

/*****
/*
/* Program for 3-phase motor
/*
/*****
void main()
{
void hinit() ;
void ainit() ;
void led_out( int ) ;
/* */
unsigned char sw, proc_no, sw_mode ;
signed      int  i, speed ;
/**** Initialization processing *****/
    hinit() ;                               /* Hardware initialization */
    ainit() ;                               /* Initialization of area used */
    __EI() ;                               /* Interrupt enabled */
    while ( stop_flag == OFF ) ;           /* Wait for origin detection */
    for ( i = 40000 ; i != 0 ; i-- ) ;
    __DI() ;                               /* Interrupt disabled */
    ainit() ;                               /* Re-initialization of area to be used */
    init_flag = OFF ;                      /* Clears (to 0) initialization flag */
    POER0 = 0x3f ;                          /* All phases active */
    proc_no = 0 ;                           /* Initialization of current processing No. */
    __EI() ;                               /* Interrupt enabled */

    while( 1 ) {
/**** Read SW *****/
        sw = ~P4 & 0x03 ;
        if ( sw == 1 ) {
            sw_mode = CW ;
        } else if ( sw == 2 ) {
            sw_mode = CCW ;
        } else {
            sw_mode = STOP ;
        }
        speed = ( SPEED_MAX * volume / 1024 ) + 1 ;
        switch( proc_no ) {
/**** Stop processing *****/
        case 0 :
            if ( sw_mode == CW ) {
                __DI() ;
                stop_flag = OFF ;           /* Clears (to 0) stop flag */
                o_speed = speed ;          /* Sets target speed */
                cont_mode = SPEED ;       /* Sets speed control mode */
                __EI() ;
                proc_no = 1 ;             /* Sets clockwise processing No. */
            } else if ( sw_mode == CCW ) {
                __DI() ;
                stop_flag = OFF ;           /* Clears (to 0) stop flag */
                o_speed = -speed ;        /* Sets target speed */
                cont_mode = SPEED ;       /* Sets speed control mode */
                __EI() ;
                proc_no = 2 ;             /* Sets counter-clockwise processing No. */
            }
            led_out( o_position - now_position ) ;
            break ;

```



```

/**** Clockwise processing *****/
case 1 :
    o_speed = speed ;                               /* Sets target speed */
    if ( (sw_mode == CCW) || (sw_mode == STOP) ) {
        o_speed = 0 ;                               /* Clears (to 0) target speed */
        stop_req = ON ;                             /* Sets (to 1) stop request flag */
        proc_no = 3 ;                               /* Sets processing No. during deceleration */
    }
    led_out( abs(o_speed) - abs(now_speed) ) ;
    break ;
/**** Counter-clockwise processing *****/
case 2 :
    o_speed = -speed                               /* Sets target speed */
    if ( (sw_mode == CW) || (sw_mode == STOP) ) {
        o_speed = 0 ;                               /* Clears (to 0) target speed
        stop_req = ON ;                             /* Sets (to 1) stop request flag */
        proc_no = 3 ;                               /* Sets processing No. during deceleration */
    }
    led_out( abs(o_speed) - abs(now_speed) ) ;
    break ;
/**** Processing during deceleration *****/
case 3 :
    if ( stop_flag == ON ) {
        timer_count = 10 ;                         /* Sets detect time as 100 ms */
        proc_no = 4 ;                               /* Sets detection wait processing No. */
    }
    led_out( abs(o_speed) - abs(now_speed) ) ;
    break ;
/**** Detection wait processing *****/
case 4 :
    if ( timer_count == 0 ) {
        __DI() ;
        cont_mode = POSITION ;                       /* Sets position control mode */
        stop_req = OFF ;                            /* Clears (to 0) stop request flag */
        stop_flag = OFF ;                           /* Clears (to 0) stop flag */
        proc_no = 0 ;                               /* Sets stop processing No. */
        o_position = now_position ;
        __EI() ;
    }
    led_out( abs(o_speed) - abs(now_speed) ) ;
    break ;
}
/**** Error processing *****/
if ( error_flag ) {
    while( 1 ) {
        led_out( 0x0000 ) ;
        timer_count = 50 ;
        while( timer_count ) ;
        led_out( error_flag ) ;
        timer_count = 50 ;
        while( timer_count ) ;
    }
}
}

```

8.4 Motor Control Interrupt Servicing (0.4 ms Interval)

```

/*****
/*
/*  TM4 interrupt servicing (0.4 ms Interval)
/*
/*
/*****
__interrupt
void int_cm4(void)
{
int sin(int) ;
/* */
signed short    iu, iv ;
signed short    now_enc, sa_enc, sa_position ;
signed int      wrm, wre, trm, tre ;
signed int      d_speed, o_iqp, o_iq, id, iq ;
signed int      o_vd0, o_vq0, o_vd, o_vq ;
signed int      o_vu, o_vv, o_vw ;
/* */
    ADSCM00 = 0x9800 ;          /* Starts A/D conversion for u-phase current */
    ADSCM10 = 0x9800 ;          /* Starts A/D conversion for v-phase current */
/**** Encoder processing *****/
    now_enc = -TM10 ;
    sa_enc = now_enc - before_enc ;
    if ( abs( sa_enc ) > ( MAXPULSE / 2 ) ) {
        if ( sa_enc < 0 ) {
            sa_enc += MAXPULSE ;
        } else {
            sa_enc -= MAXPULSE ;
        }
    }
    before_enc = now_enc ;
    wrm = now_speed = sa_enc ;          /* Sets current speed */
    wre = wrm * P ;
    now_position += sa_enc ;           /* Updates current position */
    if ( now_enc < 0 ) {
        now_enc += MAXPULSE ;
    }
    trm = now_enc ;
    tre = ( trm * P + OFFSET ) % MAXPULSE ;
    if ( (stop_req == ON) && (abs(sa_enc) == 0) ) {
        stop_flag = ON ;              /* Sets (to 1) the stop flag */
    }
/**** Target speed and position update processing *****/
    if ( cont_mode == SPEED ) {
        o_position += o_speed ;
    }
    sa_position = o_position - now_position ;
    if ( init_flag == OFF ) {
        if ( abs(sa_position) > SA_POSI_MAX ) {
            POERO = 0 ;                /* PWM output off */
            error_flag = ERR_NO2 ;     /* Sets (to 1) position error */
        }
    }
    if ( cont_mode == POSITION ) {
        sa_position = ( sa_position * kp ) >> KPGETA ;
    }
}

```

```

/**** Speed control processing *****/
d_speed = sa_position - wrm ;
o_iqp = ksp * d_speed ;
o_iq = ( o_iqp + i_speed ) >> KSIGETA ;
if ( i_speed > IS_MAX ) {
    i_speed = IS_MAX ;
} else if ( i_speed < -IS_MAX ) {
    i_speed = -IS_MAX ;
} else {
    i_speed += ( ksi * d_speed ) ;
}

/**** Current control processing *****/
iu = ( ( ADCR00 & 0x3ff ) - 0x200 ) ; /* u-phase current value */
iv = ( ( ADCR10 & 0x3ff ) - 0x200 ) ; /* v-phase current value */
ADSCM00 = 0x9801 ; /* Starts A/D conversion for speed volume */
if ( ( abs(iu) > MAX_I ) || ( abs(iv) > MAX_I ) ) {
    POER0 = 0 ; /* PWM output off */
    error_flag = ERR_NO1 ; /* Sets (to 1) overcurrent error */
}
id = ( ( ( iv * sin( tre ) ) - ( iu * sin( tre + 6667 ) ) ) ) >> SGETA ;
iq = ( ( ( iv * sin( tre + 2500 ) ) - ( iu * sin( tre + 9167 ) ) ) ) >> SGETA ;
o_vd = ( ki * -id ) >> KIGETA ;
o_vq = ( ki * ( o_iq - iq ) ) >> KIGETA ;
if ( o_vd > VMAX ) {
    o_vd = VMAX ;
} else if ( o_vd < -VMAX ) {
    o_vd = -VMAX ;
}
if ( o_vq > VMAX ) {
    o_vq = VMAX ;
} else if ( o_vq < -VMAX ) {
    o_vq = -VMAX ;
}

/**** Three-phase voltage conversion processing *****/
if ( init_flag == ON ) {
    o_trm += 10 ;
    tre = ( o_trm * P ) % MAXPULSE ;
    o_vd = 0 ;
    o_vq = 25 ;
    if ( o_trm > 20000 ) { /* Checks for completion of initial two rotations */
        stop_flag = ON ;
        POER0 = 0 ; /* PWM output off */
    }
}
o_vu = ( ( ( o_vd * sin( tre + 2500 ) ) - ( o_vq * sin( tre ) ) ) ) >> SGETA ;
o_vv = ( ( ( o_vd * sin( tre + 9167 ) ) - ( o_vq * sin( tre + 6667 ) ) ) ) >> SGETA ;
o_vw = -o_vu - o_vv ;

/**** PWM conversion output processing *****/
if ( P4 & 0x08 ) {
    POER0 = 0 ; /* PWM output off */
    error_flag = ERR_NO3 ; /* Sets (to 1) drive error */
}
if ( error_flag == 0 ) {
    P4 = 0 ;
    P4 = 0x04 ; /* Watchdog timer pulse output */
}
BFCM00 = o_vu + ( BFCM_DATA / 2 ) ;
BFCM01 = o_vv + ( BFCM_DATA / 2 ) ;
BFCM02 = o_vw + ( BFCM_DATA / 2 ) ;

volume = ADCR01 ; /* Speed volume value input */
}

```

8.5 Wait Timer Interrupt Servicing (10 ms Interval)

```
/*
*****
/*
/*  CC30 interrupt servicing (10 ms interval)
/*
/*
*****
__interrupt
void int_cc30(void)
{
    if ( timer_count != 0 ) {
        timer_count -= 1 ;
    }
}
```

8.6 Peripheral I/O Initialization Processing

```

/*****
/*
/* Hardware (peripheral I/O) initialization
/*
/*
/*****
void hinit( void )
{
void led_out( int ) ;
/**** Port mode register initialization *****/
    PMC1 = 0x07 ; /* Selects encoder input */
    PMC2 = 0x00 ;
    PM2 = 0x00 ; /* Sets LED's low-order port */
    PMC3 = 0x00 ;
    PM3 = 0x00 ; /* Sets LED's high-order port */
    P4 = 0x00 ; /* Watchdog timer low output */
    PMC4 = 0x00 ;
    PM4 = 0x04 ;
    led_out( 0x0000 ) ; /* LED output OFF */
/**** Timer 3 mode setting *****/
    PRM03 = 0 ; /* fclk = fxx/2 */
    TMC30 = 0x31 ; /* 50 MHz/2/16 (0.64  $\mu$ s) */
    TMC31 = 0x09 ; /* Selects compare mode */
    CC30 = 15625 ; /* _int_cc30 10 ms interval */
    TMC30 = 0x33 ; /* Starts timer */
    CC3IC0 = 0x03 ; /* Resets cc30 interrupt mask */
/**** Timer 4 mode setting *****/
    TMC4 = 0x31 ; /* 50 MHz/2/32 (0.64  $\mu$ s) */
    CM4 = 625 ; /* _int_cm4 0.4 ms interval */
    TMC4 = 0x33 ; /* Starts timer */
    CM4IC0 = 0x02 ; /* Clears cm4 interrupt mask */
/**** Initial settings for ADC0 and ADC1 *****/
    ADSCM00 = 0x0000 ;
    ADSCM01 = 0x0000 ;
/**** Initial setting for ANI10 *****/
    ADSCM10 = 0x0000 ;
    ADSCM11 = 0x0000 ;
/**** Timer 00 (TM00) initialization *****/
    PRM01 = 0 ; /* fclk = fxx/2 */
    SPEC0 = 0x0000 ; /* TOMR0 write enabled */
    TOMR0 = 0x03 ; /* Output mode setting */
    PSTO0 = 0x00 ; /* Real-time output prohibited */
    BFCM00 = BFCM_DATA/2 ; /* Sets 50% duty */
    BFCM01 = BFCM_DATA/2 ; /* Sets 50% duty */
    BFCM02 = BFCM_DATA/2 ; /* Sets 50% duty */
    BFCM03 = CM3_DATA ; /* Sets PWM cycle */
    DTRR0 = 50 ; /* Dead time: 2  $\mu$ s */
    POER0 = 0x3f ; /* All phases active */
    TMC00 = 0x8018 ; /* Starts TM00 timer */
/**** Timer 10 (TM10) initialization *****/
    PRM02 = 0 ; /* fclk = fxx/4 */
    NRC10 = 0x03 ; /* Selects noise elimination clock */
    TUM0 = 0x80 ; /* Selects UDC mode */
    PRM10 = 0x07 ; /* Select operation mode 4 */
    SESA10 = 0x00 ; /* Selects falling edge */
    TMC10 = 0x40 ; /* Starts count */
}

```

8.7 Common Area Initialization Processing

```

/*****
/*
/* Common area initialization
/*
/*
/*****
void ainit( void )
{
    init_flag = ON ;                /* Sets (to 1) initial flag */
    stop_flag  = OFF ;              /* Clears (to 0) stop flag */
    stop_req   = OFF ;              /* Clears (to 0) stop request flag */
    error_flag = 0 ;                /* Clears (to 0) error flag */
    cont_mode  = POSITION ;          /* Sets position control as control mode */
    now_position = 0 ;              /* Sets "0" as current position */
    o_position = 0 ;                /* Sets "0" as target position */
    o_speed    = 0 ;                /* Sets "0" as target speed */
    before_enc = -TM10 ;           /* Sets previous encoder values */
    i_speed    = 0 ;                /* Set "0" as speed integral value */
    o_trm     = 0 ;                /* Clears (to 0) the target value for initialization */
}

```

8.8 LED Display Output Processing

```

/*****
/*
/* LED display sub-routine
/* data: display data
/*
/*
/*****
void led_out( int data )
{
    int    dispd ;
    /* */
    dispd = abs(data) ;
    if ( dispd >= 32767 ) {
        dispd = 32767 ;
    }
    if ( data < 0 ) {
        dispd |= 0x8000 ;
    }
    P3 = ~dispd >> 8 ;
    P2 = ~dispd ;
}

```

8.9 Calculation Processing

```

/*****
/*
/* sin x
/* data
/* pulse unit
/* Returned value
/* sign value*16384
/*
/*****
int sin( int x )
{
    x = x % MAXPULSE ;
    if ( x < 0 ) x += MAXPULSE ;
    if ( x < (MAXPULSE/4) ){
        return sins( x ) ;
    } else if ( x < (MAXPULSE/2) ) {
        return sins( (MAXPULSE/2) - x ) ;
    } else if ( x < (MAXPULSE*3/4) ) {
        return -sins( x - (MAXPULSE/2) ) ;
    } else {
        return -sins( MAXPULSE - x ) ;
    }
}
/*****
int sins( int x )
{
short    z1, z2, z3, z4, z5 ;
/* */
    if ( x <= (MAXPULSE/8) ) {
        z1 = (x << SGETA) / (MAXPULSE/8) ;
        z2 = z1 * z1 >> SGETA ;
        z3 = z1 * z2 >> SGETA ;
        z5 = z2 * z3 >> SGETA ;
        return ( (12868*z1) - (1322*z3) + (40*z5) ) >> SGETA ;
    } else {
        x = (MAXPULSE/4) - x ;
        z1 = (x << SGETA) / (MAXPULSE/8) ;
        z2 = z1 * z1 >> SGETA ;
        z4 = z2 * z2 >> SGETA ;
        return ( (268432772) - (5050*z2) + (252*z4) ) >> SGETA ;
    }
}

```

APPENDIX A INDEX

0.4 ms interval..... 97, 113	Drive circuit block 40
0.4 ms timer interrupt 39	Drive error 25, 39
10 ms interval..... 103, 115	Drive error input 37
10 ms timer interrupt 39	DTM00 to DTM02..... 53
B	DTRR0 53, 67
before_enc 87	E
BFCM_DATA 89	Encoder counter functions 73
BFCM00 to BFCM02..... 54	Encoder input 39, 50
BFCM00 to BFCM03..... 66	Equivalent circuits 43
BFCM03..... 54	Errors 25
Buffer register CM03..... 54	F
Buffer registers CM00 to CM02..... 54	Flow charts..... 91
Buffer registers CM00 to CM03..... 66	Functions in application circuit example..... 24
C	Functions of V850E/IA1 and V850E/IA2..... 51
Calculation processing 118	H
Circuit diagram 40	Hardware configuration 25
Clockwise rotation 25	I
CM000 to CM002..... 53	init_flag..... 86
CM003..... 54	Introduction 15
CM100..... 75	IS_MAX 89
CM101..... 75	K
CM3_DATA 89	ki..... 90
Common area 86, 110	KIGETA 89
Common area initialization processing 105, 117	kp 90
Compare register 003 54	KPGETA 89
Compare register 100 75	ksi 90
Compare register 101 75	KSIGETA 89
Compare registers 000 to 002..... 53	ksp..... 90
Constant definition 88	L
cont_mode 86	LED display 26
Control block 46	LED display output processing..... 106, 117
Control principles 43	LED output 36
Control system 43	LED output block..... 40
Coordinate conversion 47	M
Counter-clockwise rotation 25	Main routine 91, 111
CPU block 27	MAX_I 89
Current control 47, 49	MAXPULSE..... 88
Current value input..... 39	Memory map 27
D	
Dead-time timer reload register 0..... 53, 67	
Dead-time timers 00 to 02..... 53	
Determination of operation mode..... 56	
Determination of PWM frequency 55	

Microcontroller and microcontroller peripheral block 40
 Mode 4 76
 Motor control constants 90
 Motor control interrupt 113
 Motor control interrupt servicing 97, 113
 Motor controller 40
 Motor rotation indicator 40
 Motor specifications 47

N

NRC10 78

O

o_position 87
 o_speed 87
 o_trm 87
 OFFSET 88
 Operation 25
 Operation mode switch block 40
 Operating mode switch input 38
 Overcurrent error 25

P

P 88
 Peripheral I/O 36
 Peripheral I/O initialization processing 104, 116
 Pin assignments 29
 POER0 68
 Position control 46, 48
 Positioning error 25
 Prescaler mode register 10 80
 PRM01 61
 PRM02 77
 PRM10 80
 Program configuration 84
 Program lists 109
 PSTO0 64
 PWM conversion 47, 50
 PWM output 39
 PWM output enable register 0 68
 PWM software timing output register 0 64
 PWM timer function 51

R

Register settings (Timer 00) 61
 Register settings (Timer 10) 77

S

SA_POSI_MAX 89
 SESA10 81
 SGETA 89
 Signal edge selection register 10 81
 sin calculation processing 107
 SPEC0 61
 Speed control 46, 48
 Speed specification volume value input 39
 SPEED_MAX 89
 STOP operation 25
 stop_flag 87
 stop_req 86
 Subroutine processing for sin calculation 108
 System configuration 26

T

Three-phase voltage conversion 49
 Timer 0 clock selection register 61
 Timer 00 51, 53
 Timer 1/timer 2 clock selection register 77
 Timer 10 73, 74
 Timer 10 noise elimination time selection register 78
 Timer control register 00 69
 Timer control register 10 83
 Timer output mode register 0 62
 Timer unit control register 00 72
 Timer unit mode register 0 79
 TM00 51, 53
 TM10 73, 74
 TMC00 69
 TMC10 83
 TOMR write enable register 0 61
 TOMR0 62
 TUC00 72
 TUM0 79

U

UDC mode 74
 UDC mode A 74, 76
 Up/down counter mode 74
 Use of encoder counter in application circuit example 76
 Use of PWM timer in application circuit example 55

V

VMAX..... 89
volume..... 87

W

Wait timer interrupt servicing..... 103, 115
Watchdog timer circuit block 40
Watchdog timer output 38



APPENDIX B REVISION HISTORY

The history of revisions up until now is shown below. The “Applied to” column indicates the chapters in each edition.

Edition	Applied to	Contents
2nd	Throughout	<ul style="list-style-type: none"> • For V850E/IA1, the following product has been deleted: <i>μ</i>PD703117 • For V850E/IA1, the following products have been added: <i>μ</i>PD703116, <i>μ</i>PD703116(A), <i>μ</i>PD703116(A1), <i>μ</i>PD70F3116, <i>μ</i>PD70F3116(A), <i>μ</i>PD70F3116(A1) • The following products (V850E/IA2) have been added: <i>μ</i>PD703114, <i>μ</i>PD70F3114 • The status of the following product has changed from under development to development complete: <i>μ</i>PD70F3116 • Bits defined as reserved words in the device file have been specified (bits whose bit numbers are in angle brackets < >).
	CHAPTER 1 INTRODUCTION	Addition of Differences Between V850E/IA1 and V850E/IA2
		Addition of Pin configuration (top view)
		Addition of Internal block diagram
	CHAPTER 5 V850E/IA1, V850E/IA2 FUNCTIONS	Addition of cautions to Timer Unit Control Register 00 (TUC00)
		Modification of Block Diagram of Timer 10 (TM10)
		Modification of setting values in Timer 1/timer 2 clock selection register (PRM02) settings
		Modification of the description on Signal Edge Selection Register 10 (SESA 10)
		Addition of cautions to Timer Control Register 10 (TMC10)
	CHAPTER 6 PROGRAM CONFIGURATION	Modification of values in List of Constants

Facsimile Message

Although NEC has taken all possible steps to ensure that the documentation supplied to our customers is complete, bug free and up-to-date, we readily accept that errors may occur. Despite all the care and precautions we've taken, you may encounter problems in the documentation. Please complete this form whenever you'd like to report errors or suggest improvements to us.

From: _____

Name _____

Company _____

Tel. _____

FAX _____

Address _____

Thank you for your kind support.

North America

NEC Electronics Inc.
Corporate Communications Dept.
Fax: +1-800-729-9288
+1-408-588-6130

Hong Kong, Philippines, Oceania

NEC Electronics Hong Kong Ltd.
Fax: +852-2886-9022/9044

Taiwan

NEC Electronics Taiwan Ltd.
Fax: +886-2-2719-5951

Europe

NEC Electronics (Europe) GmbH
Market Communication Dept.
Fax: +49-211-6503-274

Korea

NEC Electronics Hong Kong Ltd.
Seoul Branch
Fax: +82-2-528-4411

Asian Nations except Philippines

NEC Electronics Singapore Pte. Ltd.
Fax: +65-250-3583

South America

NEC do Brasil S.A.
Fax: +55-11-6462-6829

P.R. China

NEC Electronics Shanghai, Ltd.
Fax: +86-21-6841-1137

Japan

NEC Semiconductor Technical Hotline
Fax: +81- 44-435-9608

I would like to report the following error/make the following suggestion:

Document title: _____

Document number: _____ Page number: _____

If possible, please fax the referenced page or drawing.

Document Rating	Excellent	Good	Acceptable	Poor
Clarity	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Technical Accuracy	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Organization	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>