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.

RENESAS

Application Note

78K0R/Kx3-L

Low Power Consumption Operation

This document describes how to reduce power consumption in general and how to set up power reduction for the 78K0R/Kx3-L microcontroller. The included sample program shows an example of software used to reduce power consumption in STOP mode, HALT mode, sub-HALT mode, and during normal operation.

Target device
78K0R/KE3-L microcontroller

Document No. U19552EJ1V0AN00 (1st edition)
Date Published March 2009 N

CONTENTS

- **The information in this document is current as of November, 2008. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC Electronics data sheets or data books, etc., for the most up-to-date specifications of NEC Electronics products. Not all products and/or types are available in every country. Please check with an NEC Electronics 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 the prior written consent of NEC Electronics. NEC Electronics assumes no responsibility for any errors that may appear in this document.
- NEC 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 NEC Electronics 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 Electronics 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 a customer's equipment shall be done under the full responsibility of the customer. NEC Electronics assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.
- While NEC Electronics endeavors to enhance the quality, reliability and safety of NEC Electronics 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 Electronics products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.
- NEC Electronics products are classified into the following three quality grades: "Standard", "Special" and "Specific".

  The "Specific" quality grade applies only to NEC Electronics products developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of an NEC Electronics product depend on its quality grade, as indicated below. Customers must check the quality grade of each NEC Electronics 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 Electronics products is "Standard" unless otherwise expressly specified in NEC Electronics data sheets or data books, etc. If customers wish to use NEC Electronics products in applications not intended by NEC Electronics, they must contact an NEC Electronics sales representative in advance to determine NEC Electronics' willingness to support a given application.

(Note)

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

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

M8E 02. 11-1

CHAPTER 1 OVERVIEW

This application note describes how to set up power reduction in microcontrollers, including a description of power reduction unique to the 78K0R/Kx3-L, and is intended to help users understand how to reduce power consumption in microcontrollers.

The sample program included in this application note is used to reduce power consumption in STOP mode, HALT mode, sub-HALT mode, real-time counter mode, and during normal operation. The operation modes can be switched by setting up an external switch.

CHAPTER 2 REDUCING POWER CONSUMPTION

This chapter describes how to reduce power consumption in general and specifically how to reduce power consumption by the 78K0R/Kx3-L microcontroller.

2.1 Reducing Power Consumption in General

The power that microcontrollers consume can be reduced by using the following general methods.

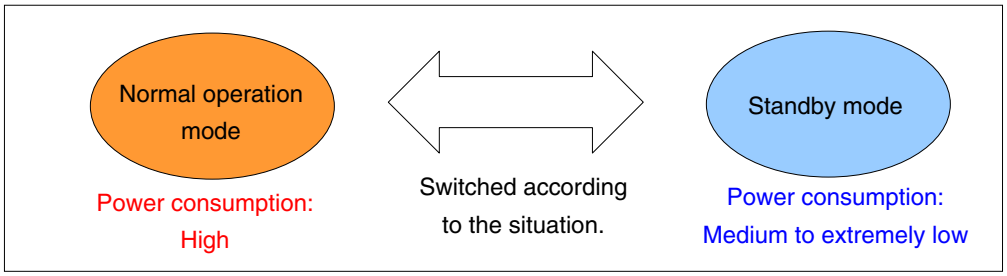
(1) Using standby mode
(2) Selecting a standby mode that is suitable for the processing
(3) Using an appropriate resonator and oscillation frequency
(4) Processing unused ports

(1) Using standby mode

A microcontroller operates mainly in normal operation mode when it performs normal processing or in standby mode to which it transitions when it does not have to perform processing.

In many applications, the microcontroller often does not have to operate or perform processing, such as when it waits for external input or waits for time to elapse by using a timer. If the microcontroller operates normally during these periods, it wastes power. However, the total power consumption can be reduced by switching between normal operation mode and standby mode according to the situation as shown in Figure 2-1.

Figure 2-1. Basic Microcontroller Operation

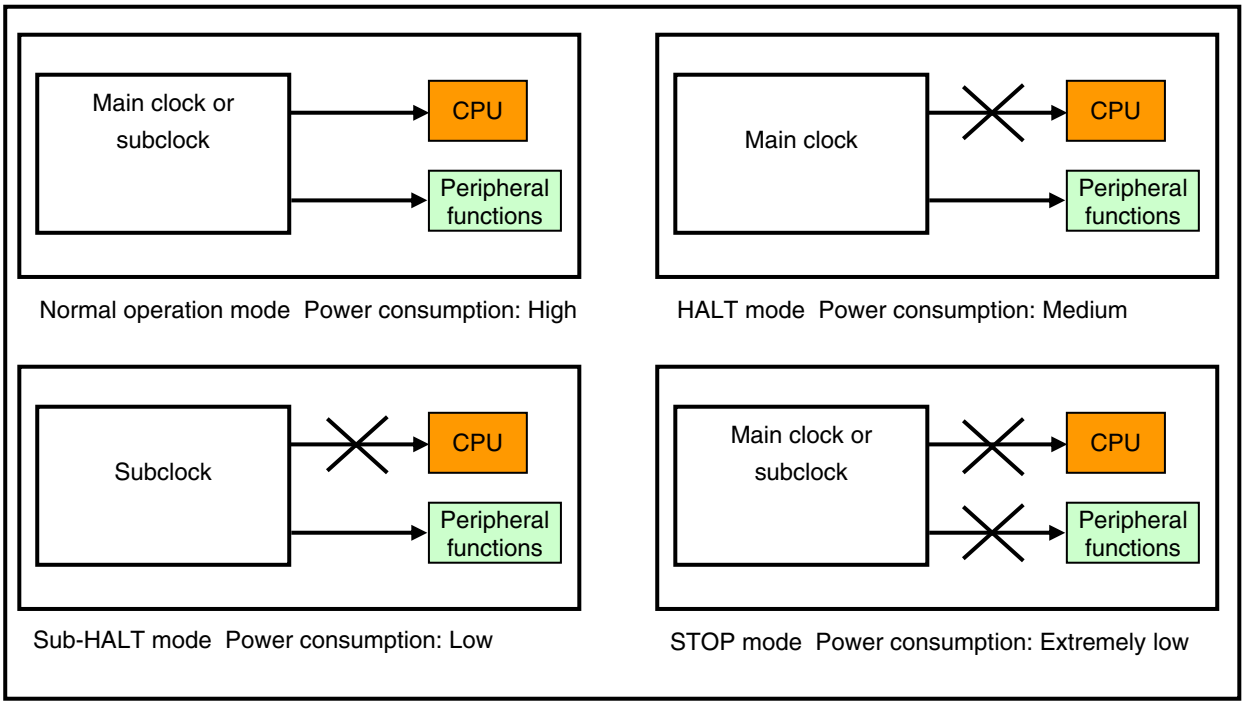


The 78K0R/Kx3-L microcontroller has the following three standby modes:

<1> HALT mode (Stops the CPU clock. Power consumption: Medium)
<2> Sub-HALT mode (Transitions to HALT mode when the CPU operates on the subsystem clock. Power consumption: Low)
<3> STOP mode (Stops clock oscillation. Power consumption: Extremely low)

Figure 2-2 shows the status in normal operation mode and the standby modes.

Figure 2-2. Status in Each Operation Mode



<1> HALT mode

In HALT mode, supplying the clock signal to the CPU is stopped, but the clock signal is supplied to the peripheral functions. Therefore, power consumption cannot be significantly reduced. However, HALT mode has the advantage that the peripheral functions can be kept operating and that the CPU can operate immediately after HALT mode is exited, because the main clock is not stopped.

<2> Sub-HALT mode

In sub-HALT mode, the internal high-speed oscillation clock can be stopped by transitioning to HALT mode while the CPU operates on the subsystem clock. Therefore, power consumption can be reduced more than in HALT mode. In addition, as in HALT mode, the real-time counter and timers can be kept operating because the clock signal is supplied to the peripheral functions. This mode is used in applications that operate a watch. As in HALT mode, the CPU can operate immediately after sub-HALT mode is exited. However, the processing speed is slower than in HALT mode, because the subsystem clock is used.

<3> STOP mode

In STOP mode, because main clock oscillation stops, power consumption can be reduced more than in HALT mode. However, most peripheral functions do not operate because the main clock is stopped. Specify a long enough period of time for clock oscillation to stabilize after STOP mode is exited.

The real-time counter can be used, because the subsystem clock is not stopped.

Table 2-1 summarizes the characteristics of each standby mode.

Table 2-1. Characteristics of Each Standby Mode

Standby Mode	Peripheral Functions	Power Consumption	Time Required to Return from a Standby
HALT mode	Usable	Medium	Short
Sub-HALT mode	Usable[Note]	Low	Short
STOP mode	Mostly stopped	Extremely low	Long

Note Peripheral functions that cannot operate on the subsystem clock cannot be used in sub-HALT mode.

(2) Selecting a standby mode that is suitable for the processing

There are three standby modes: STOP mode, HALT mode, and sub-HALT mode.
It is important to select a standby mode that is suitable for the processing as described below.

(a) Applications that have a long standby time

For applications that stay in standby mode for a very long time and execute processing after exiting standby mode and then return to standby mode, it is important to reduce the power consumption in standby mode. HALT mode, in which the clock continues to oscillate and the clock signal is supplied to the peripheral functions, is not suitable for reducing power consumption. However, power consumption can be reduced by selecting STOP mode as the standby mode, thereby performing normal processing in a short time by using the main system clock (the high-speed system clock or internal high-speed oscillation clock), and then entering STOP mode again.

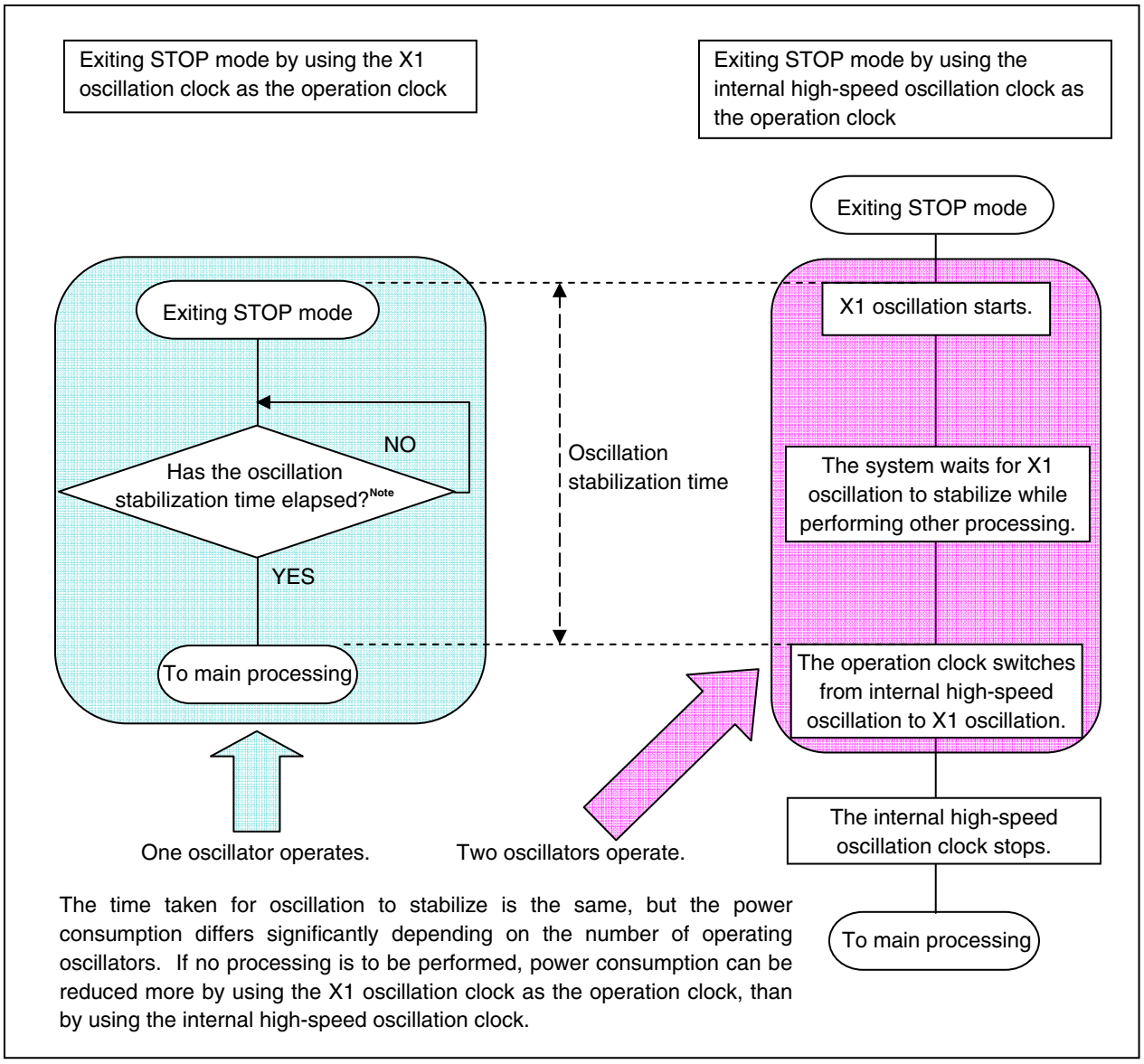
(b) Applications that perform processing by periodically returning from standby mode

For applications that perform processing by periodically returning from standby mode, if STOP mode is selected as the standby mode, the system must wait for oscillation to stabilize every time a standby is exited. Power consumption differs depending on the processing performed while the system waits for oscillation to stabilize. Figure 2-3 shows examples of processing performed during this time.
When the internal high-speed oscillation clock is used, the oscillation stabilization time is very short. When an external clock input is used, there is no oscillation stabilization time. Therefore, the amount by which power consumption that can be reduced by using STOP mode is larger than the amount of power consumed while the system waits for oscillation to stabilize.
Unless high frequency accuracy is required, such as for communication with an external device, temporarily switching from operation using the internal high-speed oscillation clock or an external clock input to operation in STOP mode is effective.

For applications that perform communication immediately after receiving a communication interrupt from an external device and then exiting standby mode, STOP mode cannot be used, because the system must wait for oscillation to stabilize.
In this case, reducing power consumption by selecting HALT mode is effective.

For applications that perform simple processing, such as periodically returning from standby mode and then checking the status, and then return to standby mode, using sub-HALT mode is effective. If only simple processing is to be performed, power consumption can be reduced by using the subsystem clock instead of the internal high-speed oscillation clock or high-speed system clock. Sub-HALT mode is used as the standby mode, because STOP mode cannot be entered when the subsystem clock is specified as the CPU clock.

Figure 2-3. Example of the Processing Performed While the System Waits for Oscillation to Stabilize



Note The system waits by using hardware, because the CPU does not operate until X1 oscillation stabilizes.

Table 2-2 shows the standby modes that are suitable for different applications. (Note that some peripheral functions cannot be used depending on the standby mode, so select the standby mode by considering the peripheral functions to use during standby.)

Table 2-2. Standby Modes Suitable for Different Applications

Application	Suitable Standby Mode
Applications that have a long standby time	STOP mode
Applications that repeatedly switch between normal operation mode and standby mode at regular times	STOP mode
Applications that repeatedly switch between normal operation mode and standby mode in a short period and have little processing	Sub-HALT mode
Applications that perform communication immediately after exiting standby mode	HALT mode

(3) Resonator and oscillation frequency to use

A crystal resonator has excellent frequency accuracy but a long oscillation stabilization time. Moreover, the longer this oscillation stabilization time, the more power is consumed, because power is also consumed while the system waits for oscillation to stabilize. For applications for which the frequency accuracy is not important, this oscillation stabilization time can be shortened by using a ceramic resonator or the internal high-speed oscillation clock. As a result, power consumption can be reduced.

The ceramic resonator has a lower frequency accuracy than a crystal resonator, but can reduce power consumption due to its short oscillation stabilization time.

The internal high-speed oscillation clock has a lower frequency accuracy than a ceramic resonator, but can reduce power consumption more than a ceramic resonator due to its very short oscillation stabilization time. By using the internal high-speed oscillation clock, no external resonator is required, which reduces costs. Table 2-3 shows the characteristics of each clock.

The oscillation frequency is also related to power consumption. In general, the higher the oscillation frequency, the higher the power consumption. If the execution speed of the application is not important, power consumption can be reduced by decreasing the oscillation frequency.

Table 2-3. Characteristics of Each Clock

	Oscillation Frequency Accuracy	Oscillation Stabilization Time
Crystal resonator	Very high accuracy (about 0.001%)	Long (from a few ms to a few dozen ms)
Ceramic resonator	Lower than a crystal resonator (about 0.5%)	Short (from a few dozen μs to a few hundred μs)
Internal high-speed oscillation clock	Lower than a ceramic resonator (about 2%)	Very short (30.7 μs (max.): 78K0R/Kx3-L)

(4) Processing unused ports

If the inputs of unused I/O ports are left open, a through current that increases power consumption is generated. This problem can be prevented to reduce power consumption by setting the port mode to output to leave them open. The generation of a through current can be prevented for input ports by using a pull-up or pull-down resistor.

2.2 Power Consumption Reduction Specific to the 78K0R/Kx3-L

This section describes how to reduce power consumption by the 78K0R/Kx3-L.

(1) Stopping the supply of the clock signal to unused peripheral functions

Specify whether to supply the clock signal to peripheral functions by setting up the peripheral enable registers (PER0, PER1, and PER2). Power consumption and noise can be reduced by stopping the supply of the clock signal to unused hardware.

Figures 2-4 to 2-6 show the formats of the peripheral enable registers and Figure 2-7 shows an overview of their functions.

Figure 2-4. Format of Peripheral Enable Register 0 (PER0)

Address: F00F0H

RTCEN	0	ADCEN	IICAEN	0	SAU0EN	0	0

SAU0EN	Control of serial array unit input clock supply
0	Stops input clock supply. • SFR used by the serial array unit cannot be written. • The serial array unit is in the reset status.
1	Supplies input clock. • SFR used by the serial array unit can be read and written.

IICAEN	Control of serial interface IICA input clock supply
0	Stops input clock supply. • SFR used by the serial interface IICA cannot be written. • The serial interface IICA is in the reset status.
1	Supplies input clock. • SFR used by the serial interface IICA can be read and written.

ADCEN	Control of A/D converter input clock supply
0	Stops input clock supply. • SFR used by the A/D converter cannot be written. • The A/D converter is in the reset status.
1	Supplies input clock. • SFR used by the A/D converter can be read and written.

RTCEN	Control of real-time counter (RTC) input clock supply[Note]
0	Stops input clock supply. • SFR used by the real-time counter (RTC) cannot be written (can be read). • Operation of the real-time counter (RTC) continues.
1	Supplies input clock. • SFR used by the real-time counter (RTC) can be read and written.

Note The input clock that can be controlled by using RTCEN is used when the register used by the real-time counter (RTC) is accessed from the CPU. RTCEN cannot control supply of the operation clock (f_{SUB}) of the real-time counter.

Caution Be sure to clear bits 6, 3, 1, and 0 to "0".

Figure 2-5. Format of Peripheral Enable Register 1 (PER1)

Address: F00F1H

0	0	0	0	OACMPEN	0	0	0

OACMPEN	Control of comparator and programmable gain amplifier input clock supply
0	Stops input clock supply. • SFR used by the comparator and programmable gain amplifier cannot be written. • The programmable gain amplifier is in the reset status.
1	Supplies input clock. • SFR used by the comparator and programmable gain amplifier can be read and written.

Caution Be sure to clear bits 7 to 4 and 2 to 0 to "0".

Figure 2-6. Format of Peripheral Enable Register 2 (PER2)

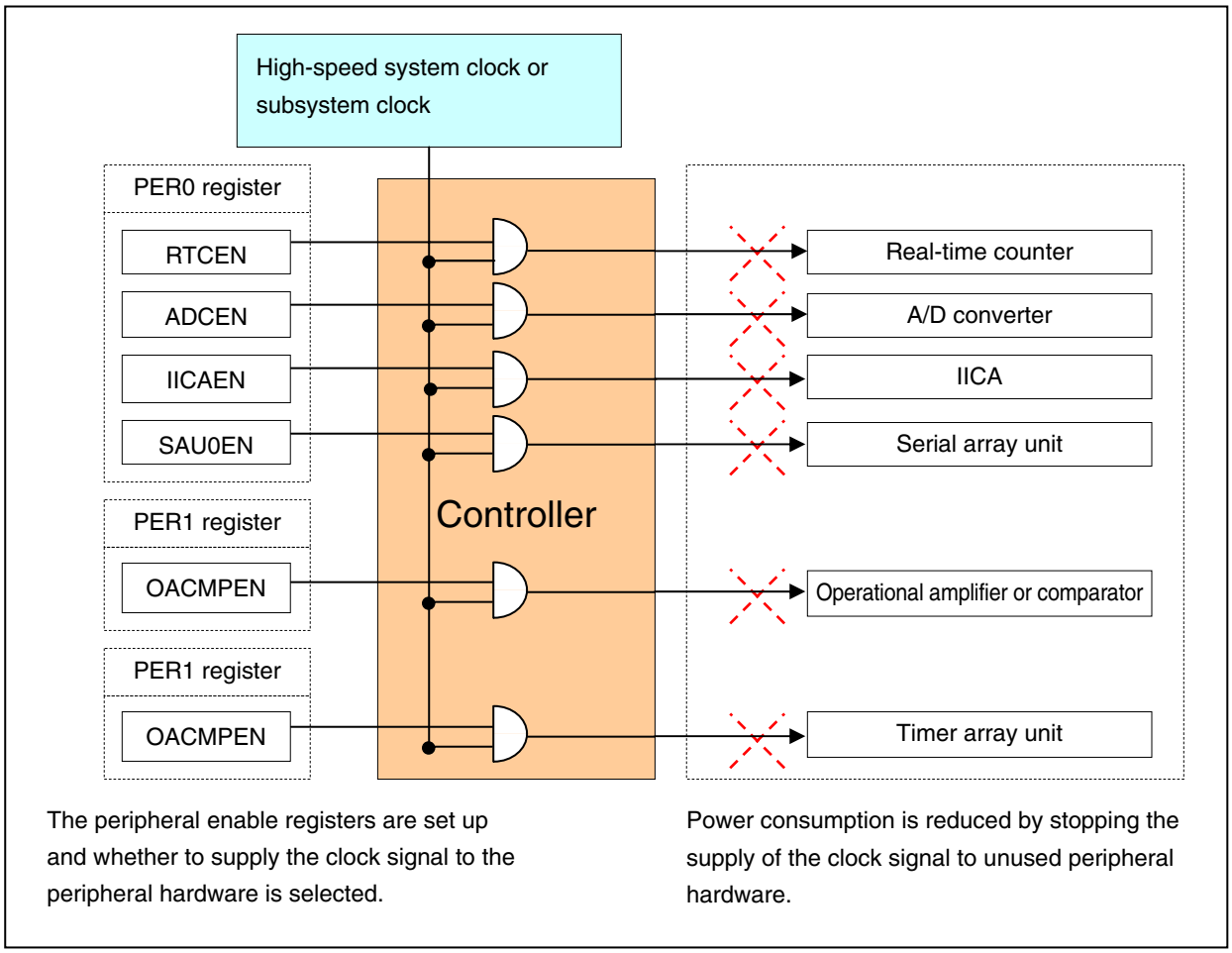
Address: F00F2H

0	0	0	0	0	0	0	TAU0EN

TAU0EN	Control of timer array unit TAUS input clock supply
0	Stops input clock supply. • SFR used by timer array unit TAUS cannot be written. • Timer array unit TAUS is in the reset status.
1	Supplies input clock. • SFR used by timer array unit TAUS can be read and written.

Caution Be sure to clear bits 7 to 1 to "0".

Figure 2-7. Functional Overview of the Peripheral Enable Registers



The peripheral enable registers are set up and whether to supply the clock signal to the peripheral hardware is selected.

Power consumption is reduced by stopping the supply of the clock signal to unused peripheral hardware.

(2) Controlling the boost circuit for using flash memory at high speeds

The boost circuit for using flash memory at high speeds is controlled by setting up the operation speed mode control register (OSMC). The values to specify differ depending on the operation clock. The power consumption at 10 MHz or less can be reduced.

By setting the RTCLPC flag of the seventh bit of OSMC to 1, the ultra-low current consumption sub-HALT mode can be entered if the conditions below are satisfied. In this mode, the power consumption of the watch can be reduced.

[Conditions for entering the ultra-low current consumption sub-HALT mode]

- The sub-HALT mode is specified.
- RTCLPC is set to 1.
- Bits other than the real-time counter of a peripheral enable register (PERn) are 0.

Figure 2-8 shows the format of the operation speed mode control register.

Figure 2-8. Format of the Operation Speed Mode Control Register (OSMC)

Address: F00F3H

RTCLPC	0	0	0	0	0	FLPC	FSEL

FLPC	FSEL	f_{CLK} frequency selection
0	0	Operates at a frequency of 10 MHz or less (default).
0	1	Operates at a frequency higher than 10 MHz.
1	0	Operates at a frequency of 1 MHz.
1	1	Setting prohibited

RTCLPC	Setting in subsystem clock HALT mode
0	Enables supply of subsystem clock to peripheral functions
1	Stops supply of subsystem clock to peripheral functions other than real-time counter

Cautions 1. Be sure to clear bits 6 to 2 to "0".

2. The operation speed mode control register can be written only once after a reset.

(3) Controlling the regulator output voltage

The regulator output voltage is controlled by setting up the regulator mode control register (RMC).

The power consumption by the 78K0R/Kx3-L can be reduced by decreasing the regulator output voltage. The RMC register can be used only under the following conditions when the regulator output voltage is fixed to low current consumption mode.

<When the X1 clock is selected as the CPU clock>

$f_X \leq 5$ MHz and $f_{CLK} \leq 1$ MHz

<When the internal high-speed oscillation clock, external input clock, or subsystem clock is selected as the CPU clock>

$f_{CLK} \leq 1$ MHz

Table 2-4 shows the conditions for the regulator output voltage.

Table 2-4. Conditions for the Regulator Output Voltage

Mode	Output Voltage	Condition
Low current consumption mode	1.8 V	While the $\overline{\text{RESET}}$ pin is reset
		During STOP mode (except during OCD mode)
		When the high-speed system clock (f_{MX}), internal high-speed oscillation clock (f_{IH}), and 20 MHz internal high-speed oscillation clock (f_{IH20}) stop while the CPU operates on the subsystem clock (f_{XT})
		When the high-speed system clock (f_{MX}), internal high-speed oscillation clock (f_{IH}), and 20 MHz internal high-speed oscillation clock (f_{IH20}) stop in HALT mode and the CPU is specified to operate on the subsystem clock (f_{XT})
Normal current mode	2.4 V	Other than the above

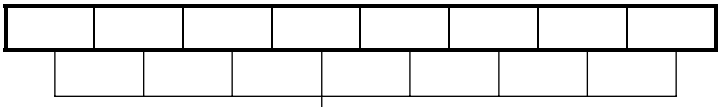
The low current consumption mode is automatically entered in the following cases:

- Upon a reset
- During standby mode (HALT, sub-HALT, or STOP)
- When the subsystem clock is used as f_{CLK} and the internal high-speed oscillation clock is stopped

Figure 2-9 shows the format of the regulator mode control register.

Figure 2-9. Format of the Regulator Mode Control Register (RMC)

Address: F00F4H



RMC[7:0]	Control of regulator output voltage
5AH	Fixed to low current consumption mode (1.8 V)
00H	Switches normal current mode (2.4 V) and low current consumption mode (1.8 V) according to the condition
Other than the above	Setting prohibited

Cautions 1. Rewrite the RMC register while the CPU operates on the subsystem clock (f_{SUB}) and the high-speed system clock (f_{MX}), internal high-speed oscillation clock (f_{IH}), and 20 MHz internal high-speed oscillation clock (f_{IH20}) are stopped.

2. The RMC register can be used only under the following conditions when the regulator output voltage is fixed to low current consumption mode.

<When the X1 clock is selected as the CPU clock>

$f_X \leq 5$ MHz and $f_{CLK} \leq 1$ MHz

<When the internal high-speed oscillation clock, external input clock, or subsystem clock is selected as the CPU clock>

$f_{CLK} \leq 1$ MHz

3. Self-programming cannot be used in low current consumption mode.

(4) 1 MHz internal high-speed oscillation mode

The 78K0R/Kx3-L has an internal high-speed oscillation clock that operates at 1 MHz. Power consumption can be reduced by setting the operation clock of the internal high-speed oscillation clock to 1 MHz by using the option bytes and setting the operation speed mode control register (OSMC) to operate at a frequency of 1 MHz. (For details about the operation speed mode control register, see **Figure 2-8**.) Figure 2-10 shows the format of the related option bytes. (For details about the format of all option bytes, see **5.1 Setting Up the Option Bytes**.)

Figure 2-10. Format of the User Option Bytes (000C1H/010C1H)

Address: 000C1H/010C1H[Note 1]

1	1	1	1	1	FRQSEL2	FRQSEL1	LVIOFF

Setting of LVI on power application

0	LVI is ON by default (LVI default start function enabled) upon reset release (upon power application)
1	LVI is OFF by default (LVI default start function stopped) upon reset release (upon power application)

Internal high-speed oscillation clock frequency

0	1	8 MHz or 20 MHz[Note 2] (default)
1	0	1 MHz[Note 3]
Other than the above		Setting prohibited

Notes 1. Specify the same value as 000C1H to 010C1H when the boot swap operation is used because 000C1H is replaced by 010C1H.

2. If 8 MHz or 20 MHz is selected, the 8 MHz internal high-speed oscillation clock automatically starts oscillating after reset release. To use the 20 MHz internal high-speed oscillation clock to operate the microcontroller, set up the 20 MHz internal high-speed oscillation control register.

The setting cannot be changed to the 1 MHz internal high-speed oscillation clock while the microcontroller operates.

3. If 1 MHz is selected, the microcontroller operates on the 1 MHz internal high-speed oscillation clock after reset release. The setting cannot be changed to the 8 MHz or 20 MHz internal high-speed oscillation clock while the microcontroller operates.

(5) Subsystem clock ultra-low power consumption oscillation mode

The 78K0R/Kx3-L has a subsystem clock oscillator whose power consumption is lower than that of conventional products. Power consumption can be further reduced by setting the oscillation mode of the XT1 oscillator to ultra-low power consumption oscillation mode by using the clock operation mode control register (CMC).

Figure 2-11 shows the format of the clock operation mode control register.

Figure 2-11. Format of the Clock Operation Mode Control Register (CMC)

Address: FFFA0H

EXCLK	OSCSEL	0	OSCSELS	0	AMPHS1	AMPHS0	AMPM

AMPH	Control of high-speed system clock oscillation frequency
0	2 MHz ≤ f_{MX} ≤ 10 MHz
1	10 MHz < f_{MX} ≤ 20 MHz

AMPHS1	AMPHS0	XT1 oscillator oscillation mode selection
0	0	Low power consumption oscillation (default)
0	1	Normal oscillation
1	x	Ultra-low power consumption oscillation

OSCSELS	Subsystem clock pin operation mode	XT1/P123 pin	XT2/P124 pin
0	Input port mode	Input port	
1	XT1 oscillation mode	Crystal resonator connection	

EXCLK	OSCSEL	High-speed system clock pin operation mode	X1/P121 pin	X2/EXCLK/P122 pin
0	0	Input port mode	Input port	
0	1	X1 oscillation mode	Crystal/ceramic resonator connection	
1	0	Input port mode	Input port	
1	1	External clock input mode	Input port	External clock input

Cautions 1. Be sure to clear bits 5 and 3 to "0".

2. Note the following because the XT1 oscillator is a circuit with low amplification in order to achieve low power consumption:

- **Pins and circuit boards include parasitic capacitance. Therefore, evaluate oscillation based on the circuit board that will actually be used to confirm that there are no problems.**
- **When using the ultra-low power consumption oscillation (AMPHS1 = 1) as the mode of the XT1 oscillator, use the recommended resonators described in the electrical specifications in the 78K0R/Kx3-L User's Manual (U19291E).**
- **Make the wiring between the XT1 and XT2 pins and the resonators as short as possible, and minimize the parasitic capacitance and wiring resistance, especially when ultra-low power consumption oscillation (AMPHS1 = 1) is selected.**
- **Create the circuit board by using material that has little wiring resistance and parasitic capacitance.**
- **Place a ground pattern that has the same potential as V_{SS} as much as possible near the XT1 oscillator.**

- **Be sure that the signal lines between the XT1 and XT2 pins, and the resonators do not cross with the other signal lines. Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **The impedance between the XT1 and XT2 pins might drop and oscillation might be disturbed due to moisture absorption of the circuit board in a high-humidity environment or dew condensation on the board. When using the circuit board in such an environment, take measures to damp-proof the circuit board, such as by coating.**
- **When coating the circuit board, use material that does not cause capacitance or leakage between the XT1 and XT2 pins.**

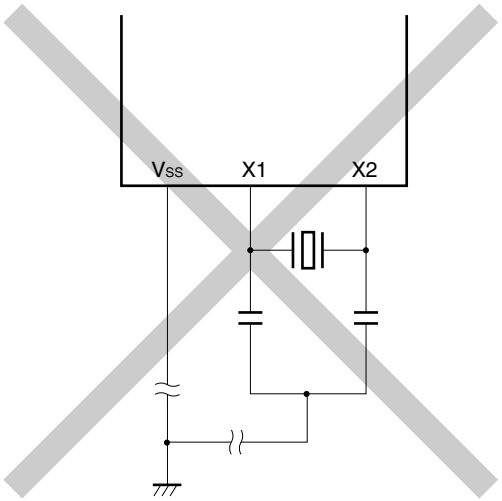
Remarks 1. f_{MX}: High-speed system clock frequency

2. x: don't care

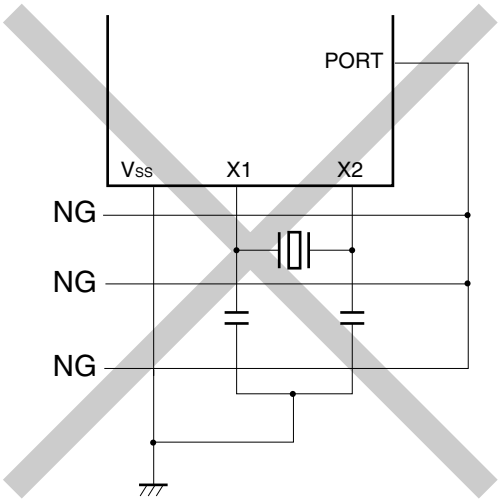
Figure 2-12 shows examples of incorrect resonator connection.

Figure 2-12. Examples of Incorrect Resonator Connection (1/2)

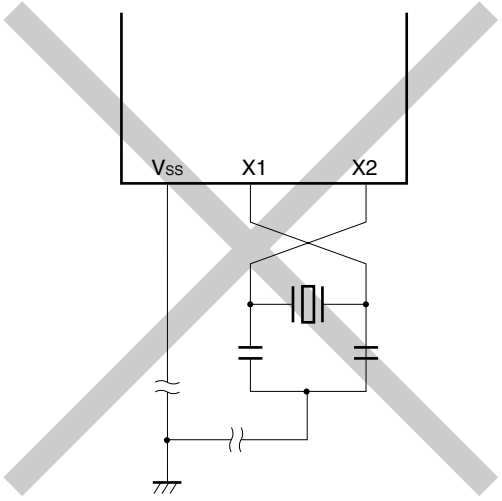
(a) Too long wiring



(b) Crossed signal line



(c) The X1 and X2 signal line wires cross.



(d) A power supply/GND pattern exists under the X1 and X2 wires.

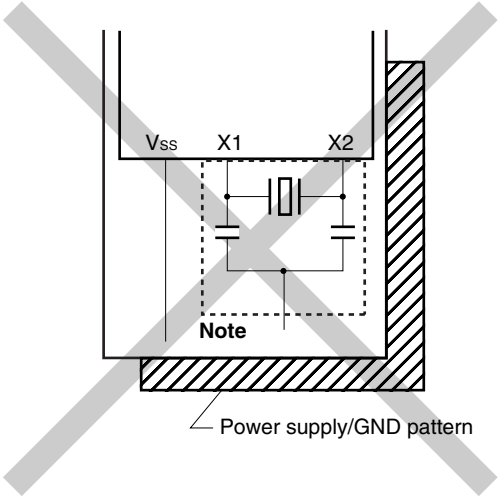
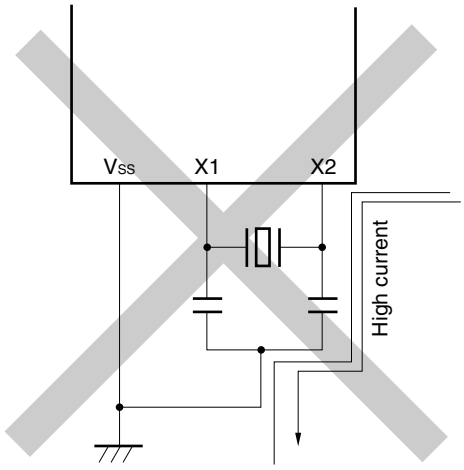
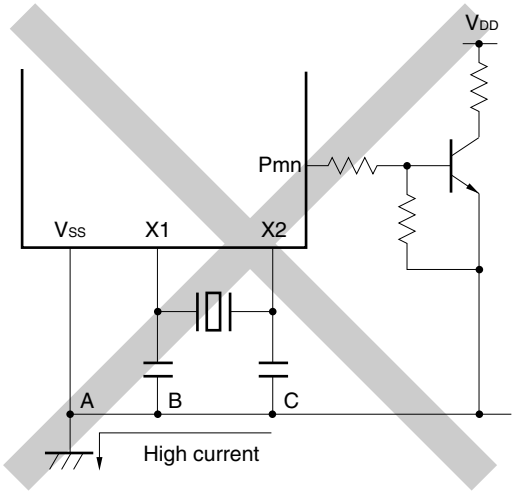


Figure 2-12. Examples of Incorrect Resonator Connection (2/2)

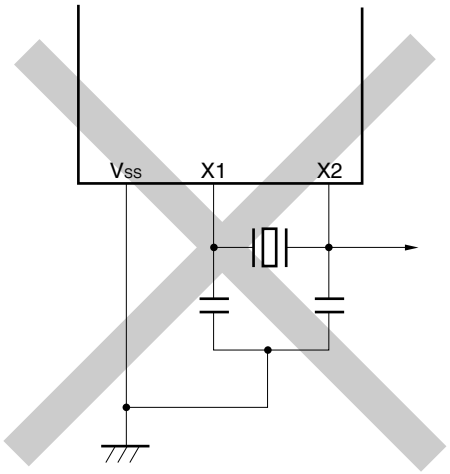
(e) Wiring near high alternating current



(f) Current flowing through ground line of oscillator (potential at points A, B, and C fluctuates)



(g) Signals are fetched



Note Do not place a power supply/GND pattern under the wiring section (section indicated by a broken line in the figure) of the X1 and X2 pins and the resonators in a multi-layer board or double-sided board.
Do not configure a layout that will cause capacitance elements and affect the oscillation characteristics.

Remark When using the subsystem clock, replace X1 and X2 with XT1 and XT2, respectively. Also, insert resistors in series on the XT2 side.

Table 2-5 shows methods for specifically reducing power consumption by the 78K0R/Kx3-L.

Table 2-5. Methods for Specifically Reducing Power Consumption by the 78K0R/Kx3-L

Method for Reducing Power Consumption	Register to Set Up
Stopping the supply of the clock signal to unused peripheral functions	Peripheral enable register (PERn)
Controlling the boost circuit for using flash memory at high speeds	Operation speed mode control register (OSMC)
Controlling the regulator output voltage	Regulator mode control register (RMC)
1 MHz internal high-speed oscillation mode	Option bytes (000C1H, 010C1H)
Subsystem clock ultra-low power consumption oscillation mode	Clock operation mode control register (CMC)

CHAPTER 3 OVERVIEW OF THE SAMPLE PROGRAM

This chapter describes the included files and used internal peripheral functions, provides an overview, and shows flowcharts of the sample program used to reduce power consumption by the 78K0R/Kx3-L.

3.1 Included Files

The compressed file to be downloaded includes the following files:

File Name	Description
main.asm (assembly language version)	Source file used for initializing the microcontroller hardware, performing main processing, intermittent STOP operation, intermittent HALT operation, and operating the RTC
main.c (C language version)	
op.asm[Note]	Assembler source file for setting up the option bytes (This file is used to specify the clock based on which the internal high-speed oscillation clock operates and the operation of the watchdog timer.)

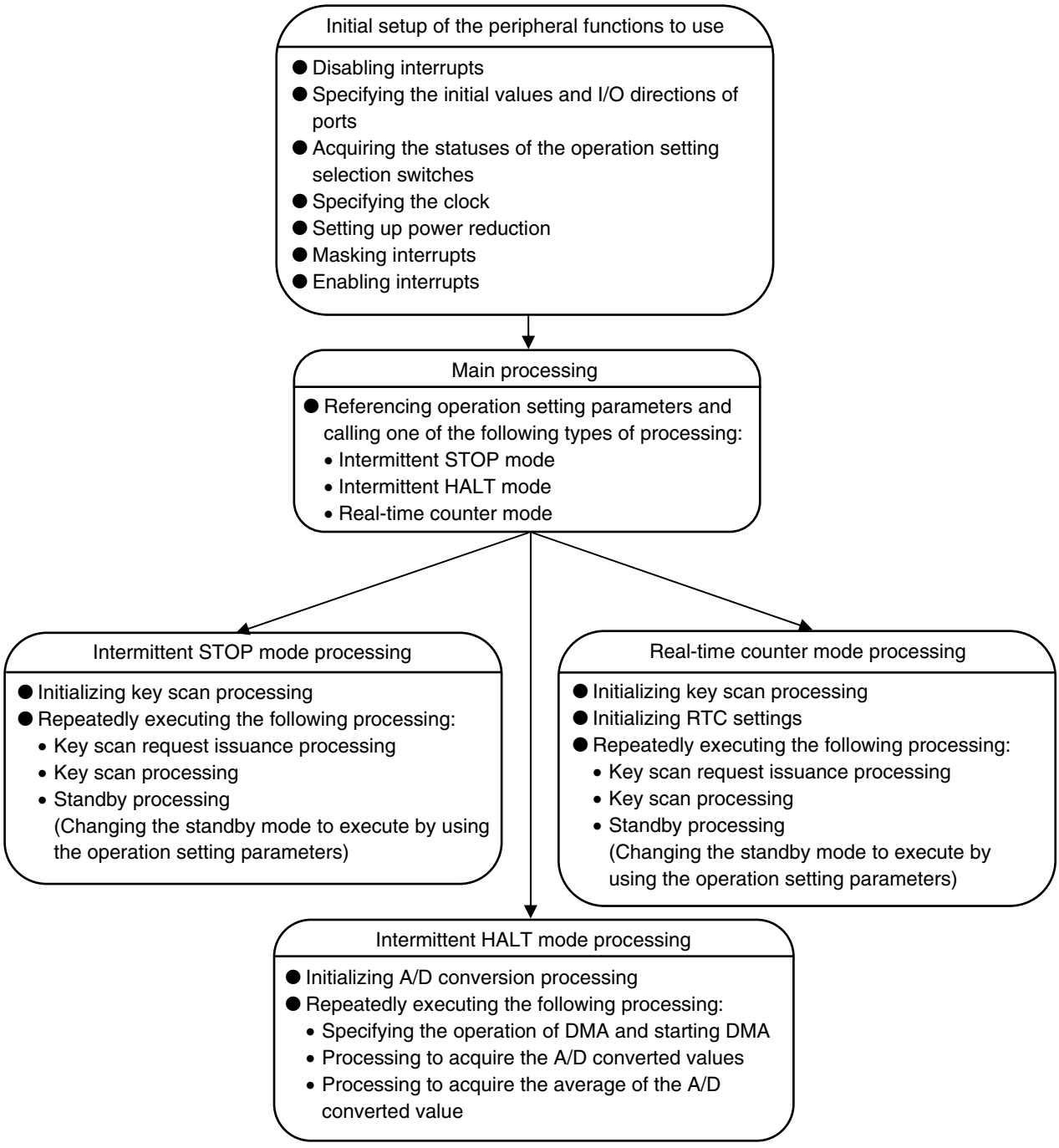
Note When using the sample program with the internal high-speed oscillation clock, the setting at the relevant section (000C1H) in op.asm must be changed to the oscillation frequency at which to use the sample program. For details, see **5.1 Setting Up the Option Bytes**.

3.2 Program Overview

In the sample program, the settings for setting up ports, acquiring the statuses of the operation setting selection switches, setting up the clock according to parameters, and reducing power consumption are initialized during the initial setup.

After these settings have been initialized, the processing branches to the main processing, and then intermittent STOP mode processing (key scan), intermittent HALT mode processing (A/D conversion), or real-time counter mode processing is called according to operation setting parameters. At the call destination, initialization is performed for each processing and then the main processing of each mode is repeatedly executed. For details, see **Figure 3-1 Status Transition Diagram of the Sample Program**.

Figure 3-1. Status Transition Diagram of the Sample Program



3.3 Used Internal Peripheral Functions

The following internal peripheral functions are used in the sample program:

- Channel 0 (TM00) of the timer array unit (TAUS):
  This is used to issue a key scan request in cycles of about 10 ms in interval timer mode.
- A/D converter:
  This is used to perform A/D conversion for the signals input to the ANI9 analog input port.
- DMA channel 0:
  This is used to transfer the A/D conversion result to the RAM area in 16-bit units.
- Real-time counter:
  This is used to execute the watch function and output a 1 Hz cycle signal for debugging.

3.4 Operation Modes

In the sample program, the operation mode, type of CPU or peripheral hardware clock, clock oscillation frequency, standby mode, and operation voltage are specified by flipping a switch.

The operation modes are described below:

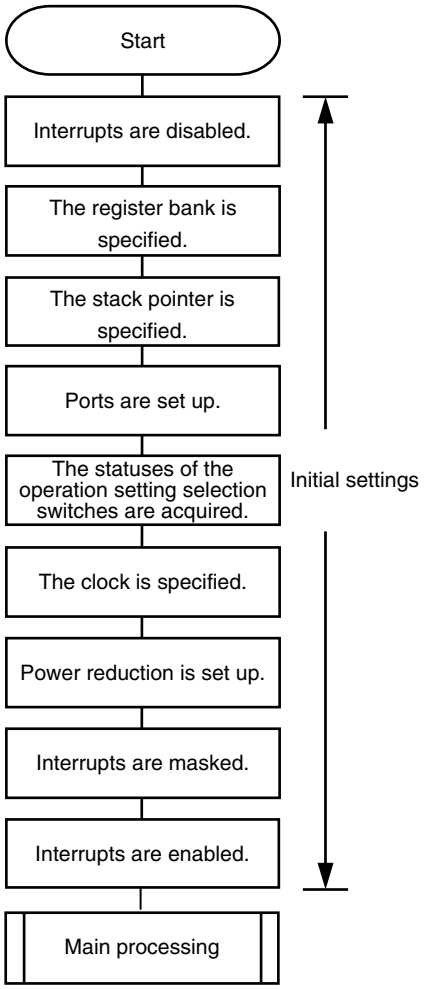
Intermittent STOP mode:	In this mode, standby mode is entered or returned from by a key scan. Power consumption can be reduced most if STOP mode is selected as the standby mode, but the operation in all modes (normal operation mode (without standby), HALT mode, sub-HALT mode, and STOP mode) can be checked to compare the power consumption in each mode.
Intermittent HALT mode:	In this mode, A/D conversion is performed a specific number of times, and then the operation returns from standby mode. Power consumption can be reduced most if HALT mode is selected as the standby mode, but the operation in normal operation mode (without standby) can be checked for comparison.
Real-time counter mode:	In this mode, standby mode is entered or returned from by a key scan. The operation of the real-time counter can be checked and power consumption can be measured in STOP mode and sub-HALT mode.

3.5 Flowcharts

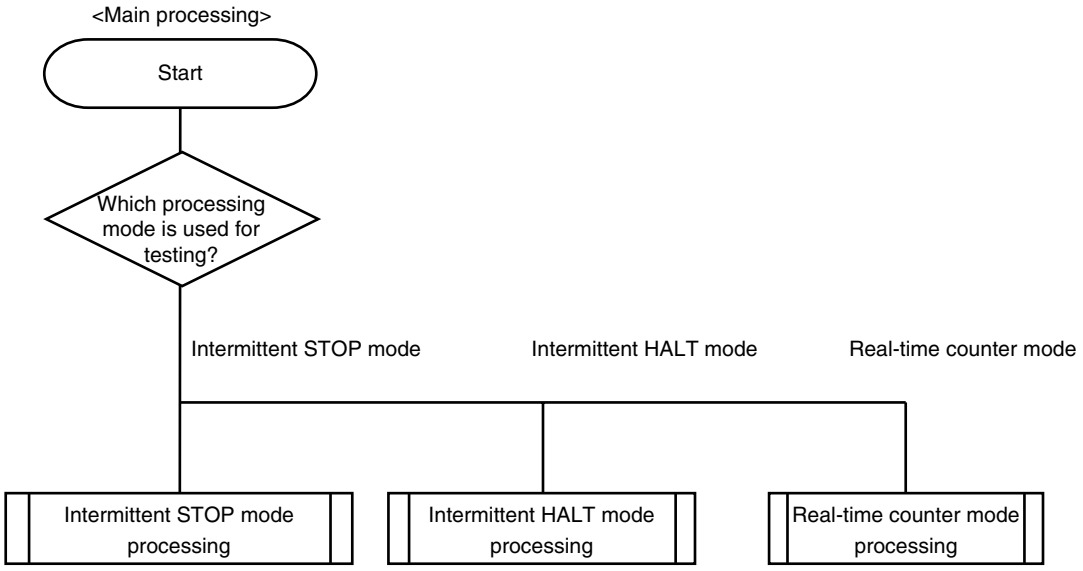
The flowcharts for the sample program are shown below.

3.5.1 Initialization processing

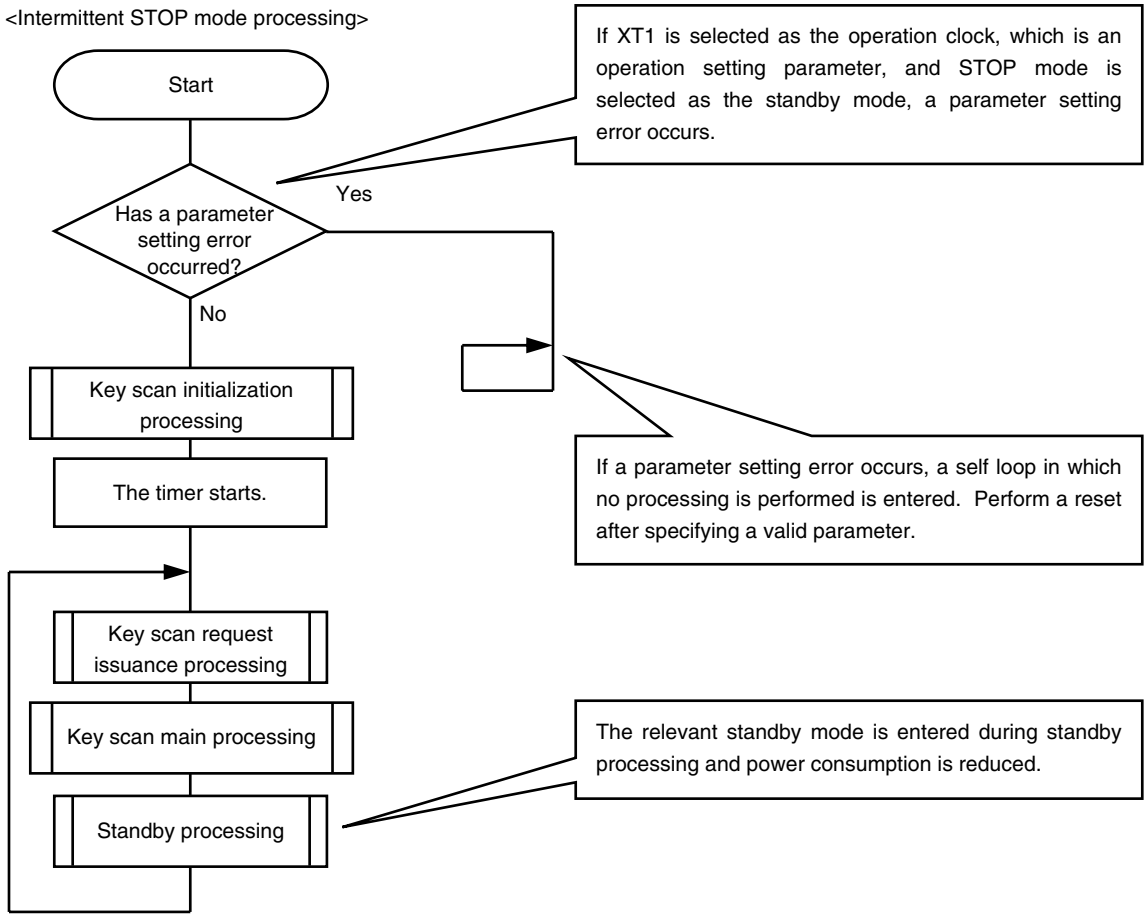
<Processing to initially set up the peripheral functions to use after a reset release>



3.5.2 Main processing

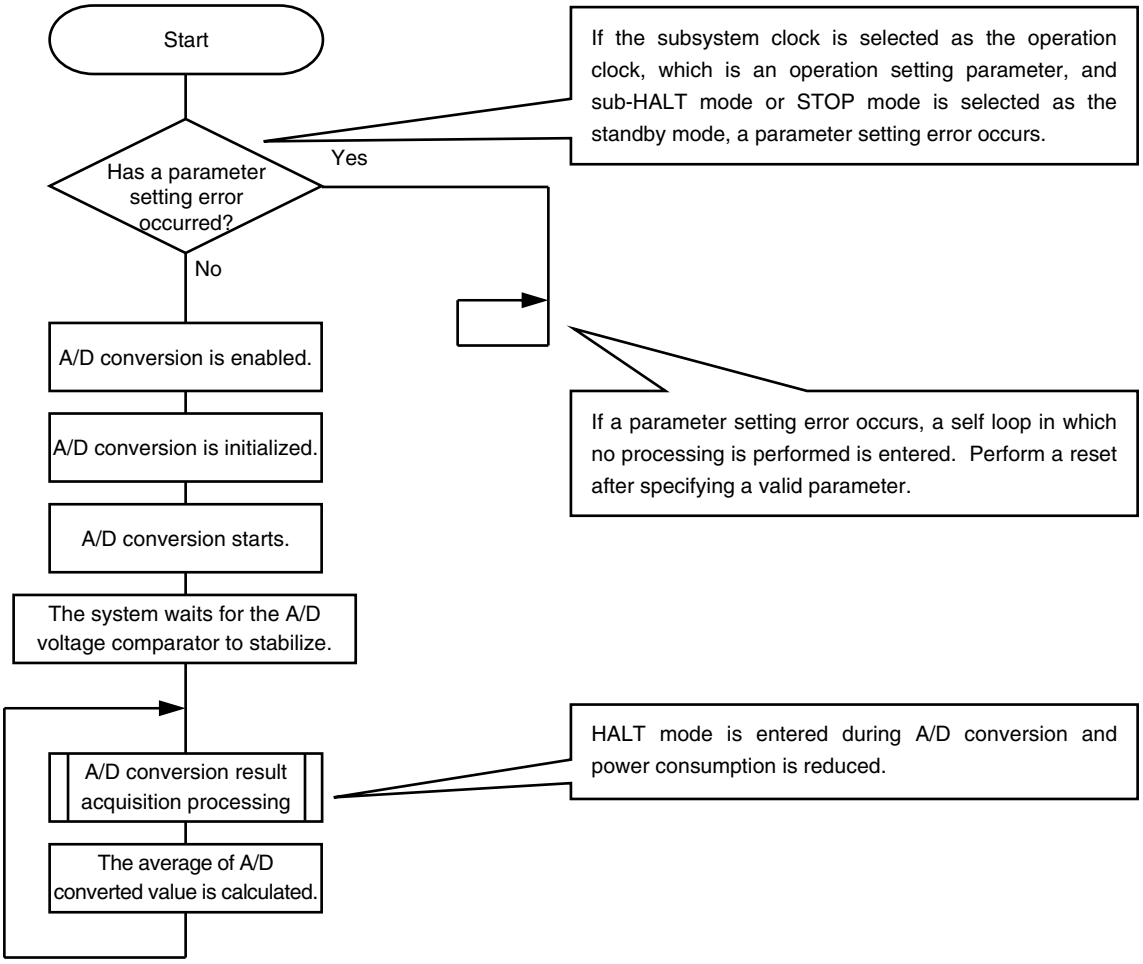


3.5.3 Intermittent STOP mode processing

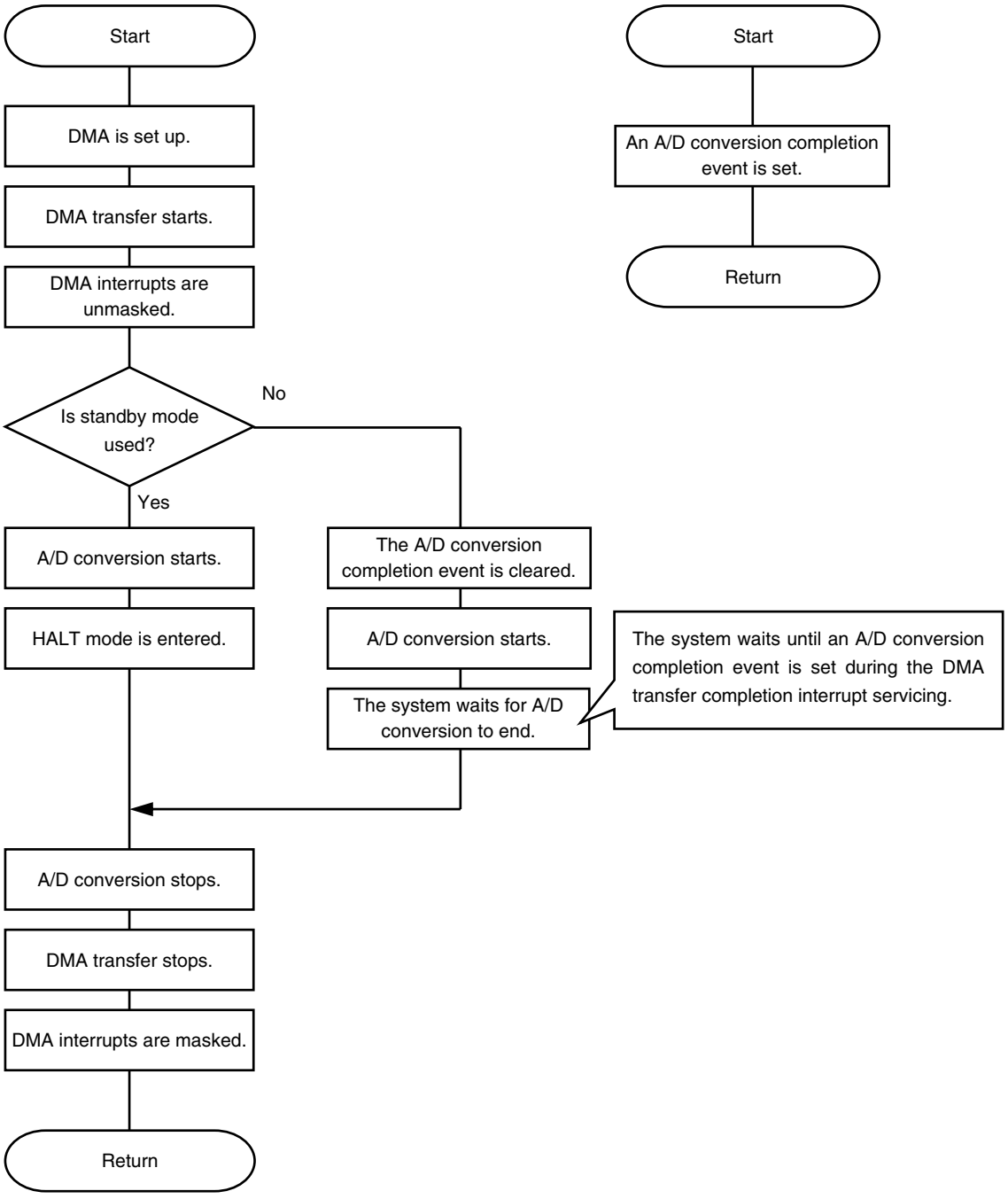


3.5.4 Intermittent HALT mode processing

<Intermittent HALT mode processing>

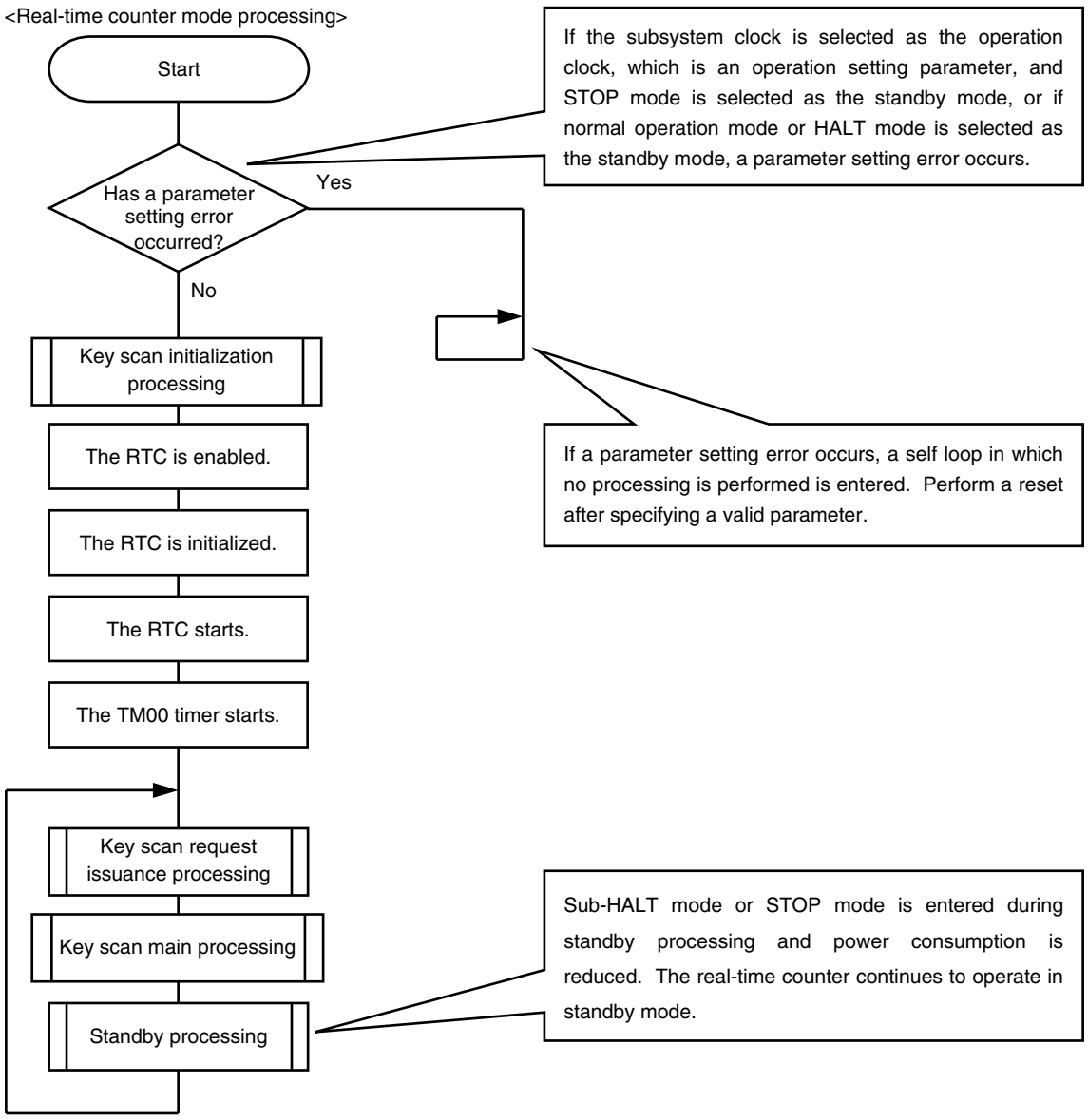


<A/D conversion result acquisition processing>



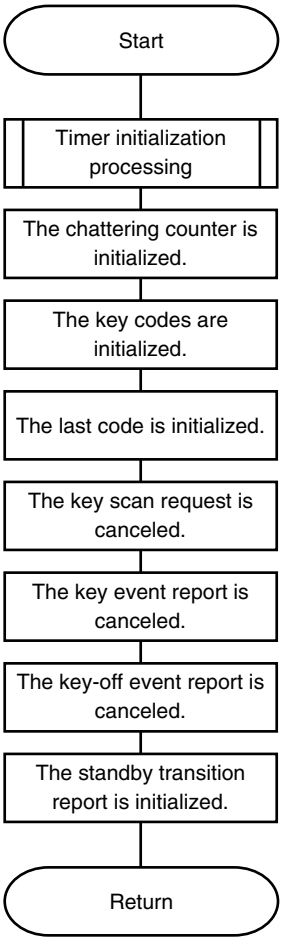
<DMA transfer completion interrupt servicing (using INTDMA0)>

3.5.5 Real-time counter mode processing

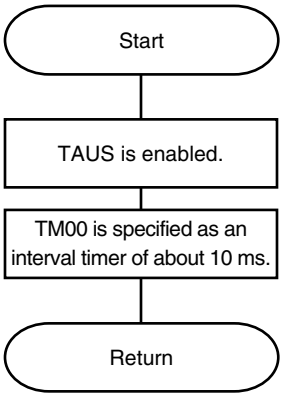


3.5.6 Processing related to key scan processing

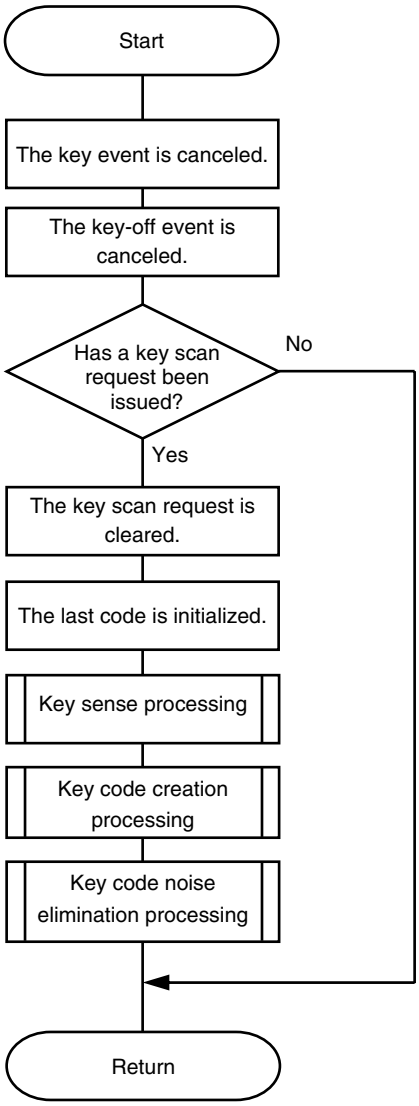
<Key scan initialization processing>



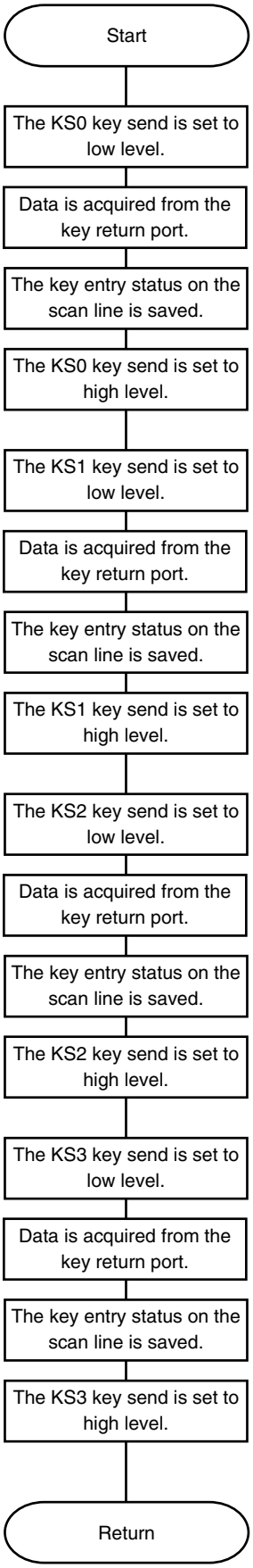
<Timer initialization processing>



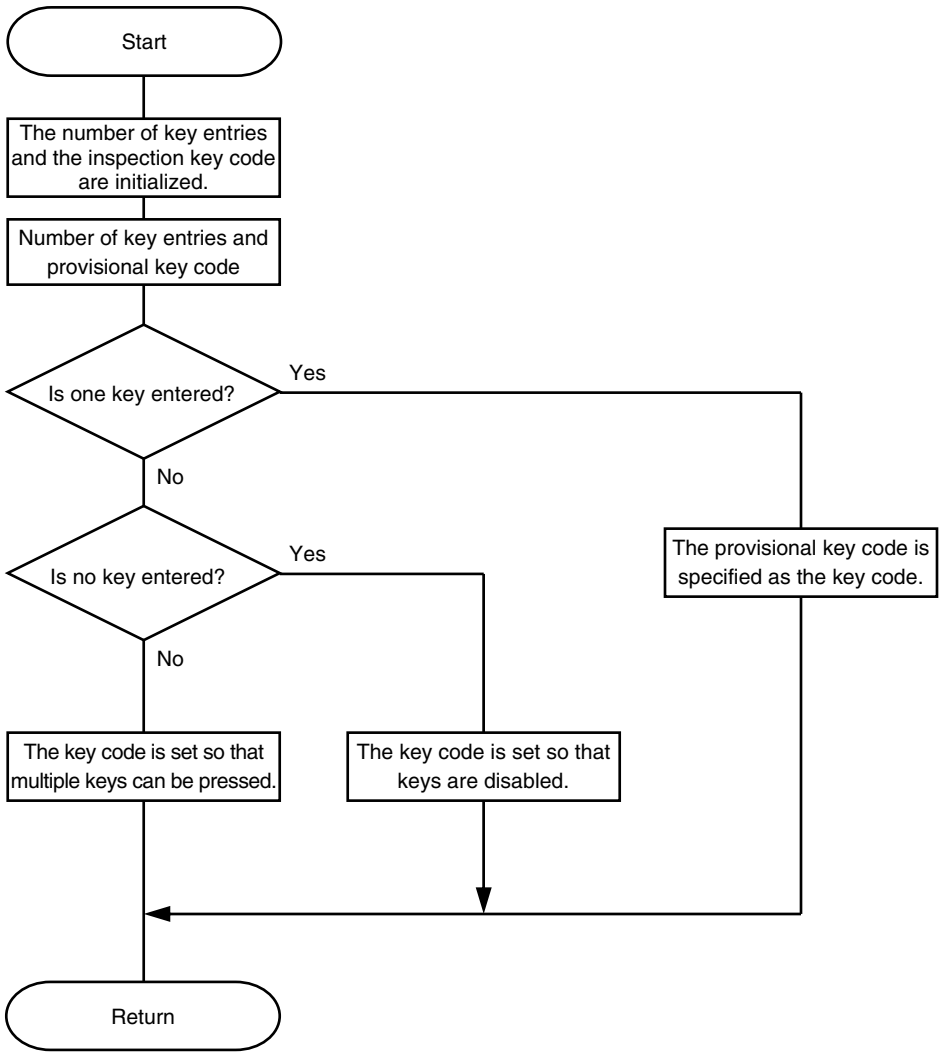
<Key scan main processing>



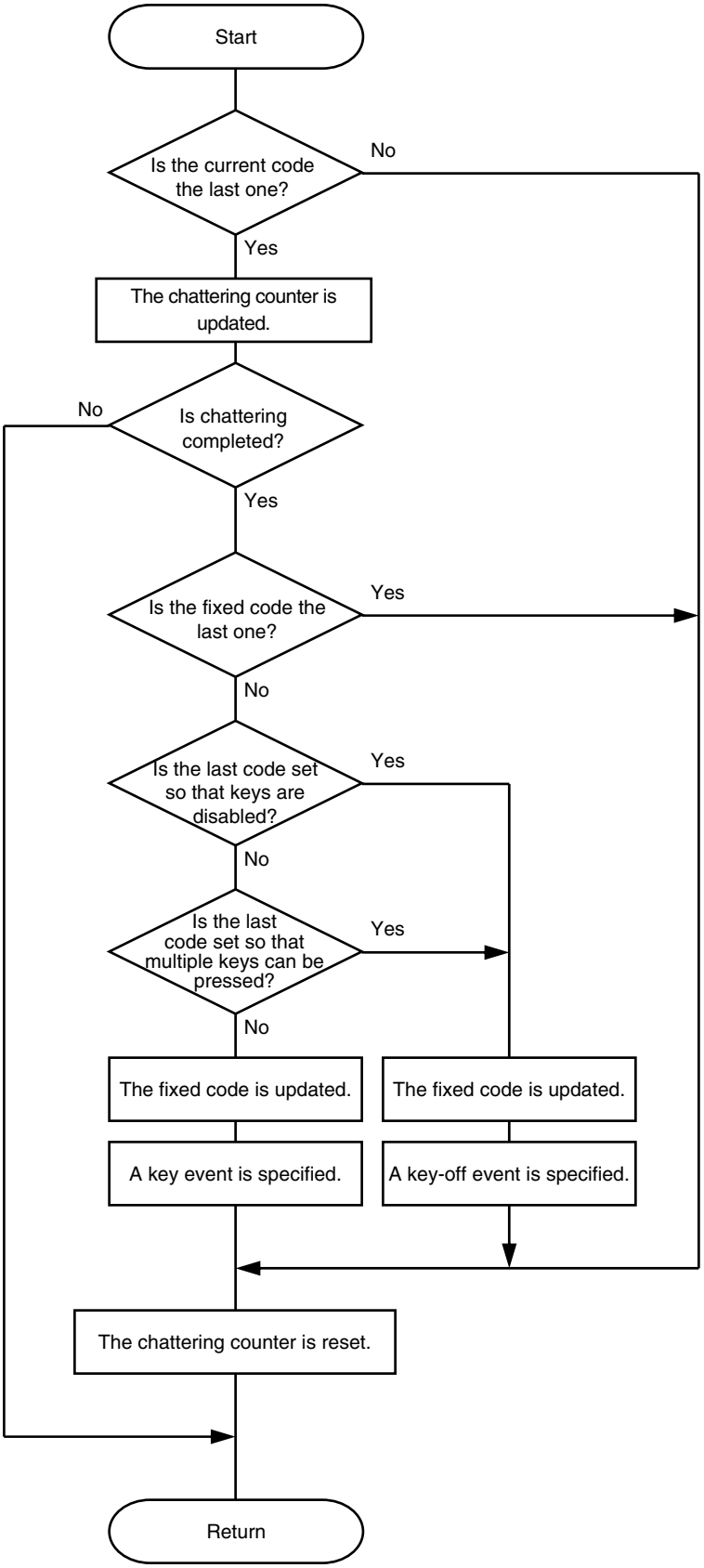
<Key sense processing>



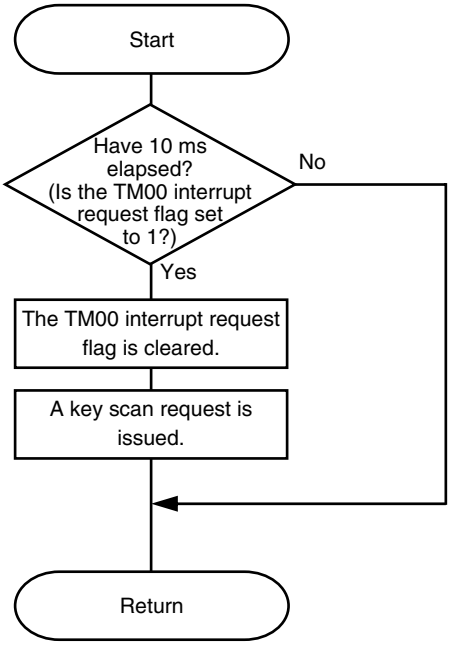
<Key code creation processing>

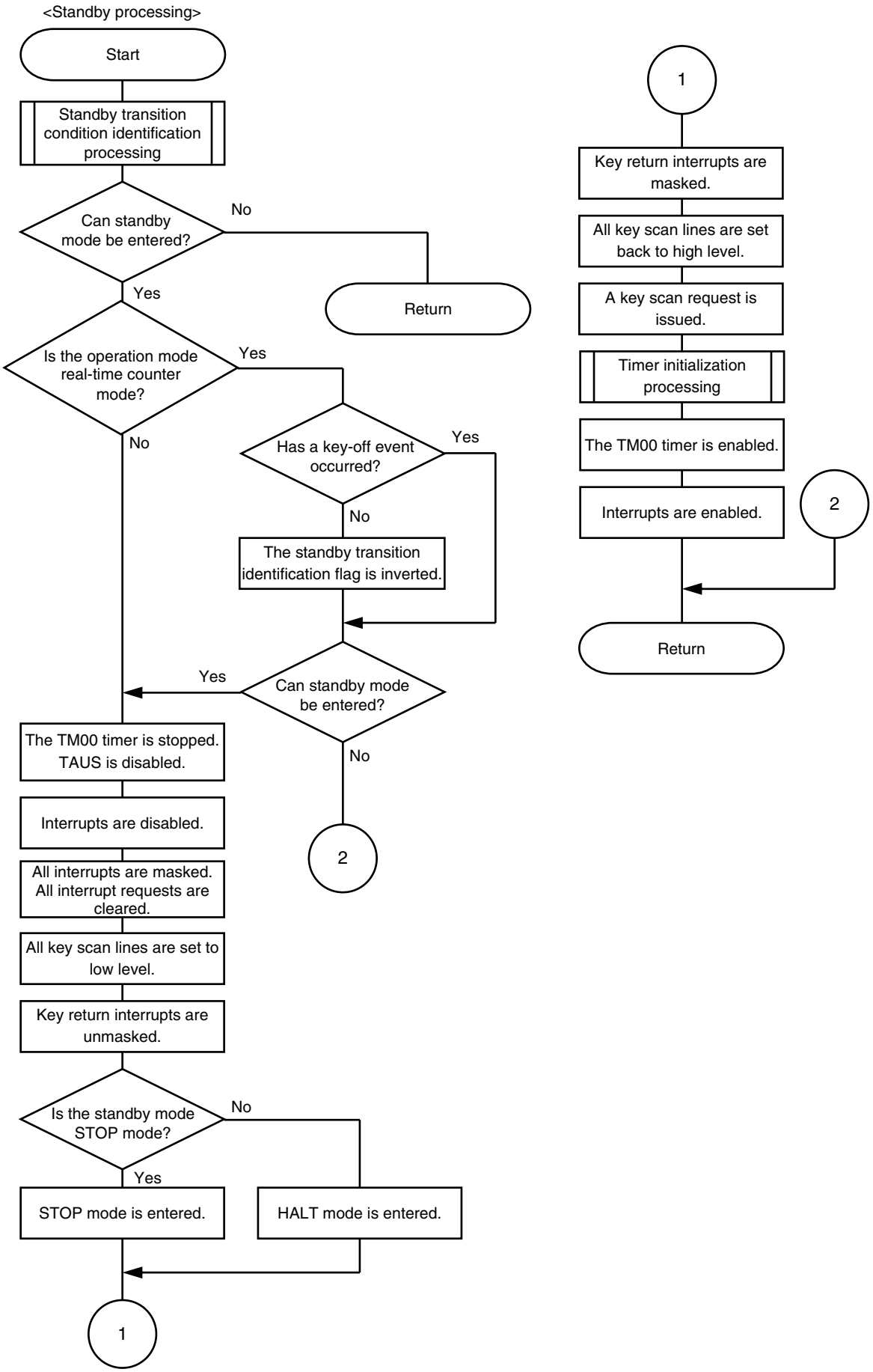


<Key code noise elimination processing>

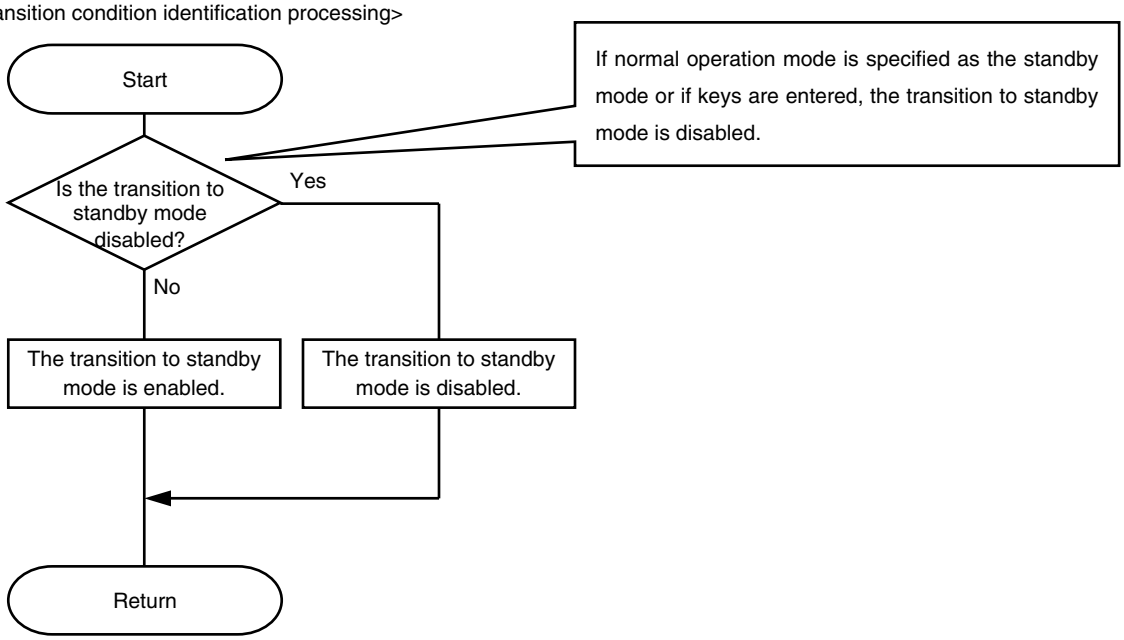


<Key scan request issuance processing>




<Standby processing>
Start
Standby transition condition identification processing
Can standby mode be entered?
No
Return
Yes
Is the operation mode real-time counter mode?
Yes
No
Has a key-off event occurred?
Yes
No
The standby transition identification flag is inverted.
Can standby mode be entered?
Yes
No
The TM00 timer is stopped. TAUS is disabled.
Interrupts are disabled.
All interrupts are masked. All interrupt requests are cleared.
All key scan lines are set to low level.
Key return interrupts are unmasked.
Is the standby mode STOP mode?
No
Yes
STOP mode is entered.
HALT mode is entered.
1
2
1
Key return interrupts are masked.
All key scan lines are set back to high level.
A key scan request is issued.
Timer initialization processing
The TM00 timer is enabled.
Interrupts are enabled.
2
Return

<Standby transition condition identification processing>



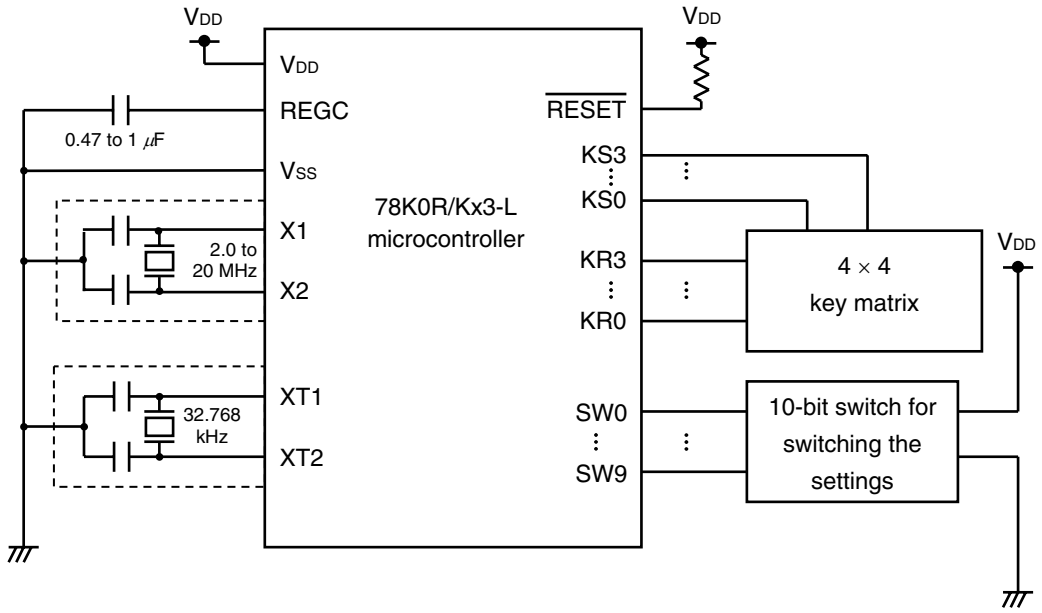
CHAPTER 4 OPERATION ENVIRONMENT

This chapter describes the environment necessary to use the sample program.

4.1 System Circuit Diagram

A system circuit diagram is shown below. Unused pins that are not shown in the figure below are set to output mode by the program.

Figure 4-1. System Circuit Diagram



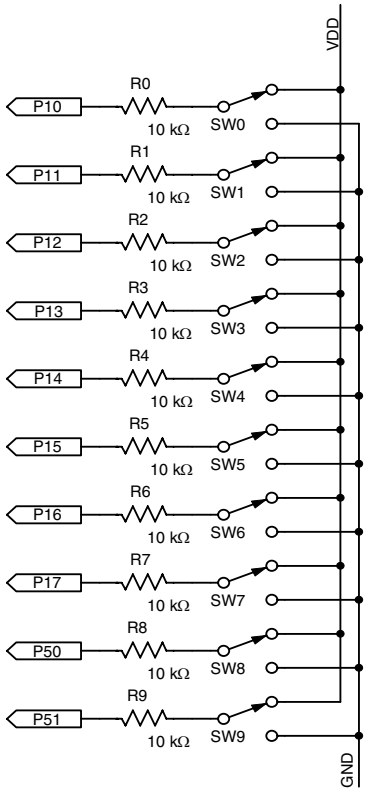
Caution The sample program cannot be used for the 78K0R/KC3-L microcontroller, because the 78K0R/KC3-L microcontroller does not have P14 to P17.

4.2 Operation Setting Selection Switches

4.2.1 Circuit diagram of the operation setting selection switches

A circuit diagram of the operation setting selection switches is shown below.

Figure 4-2. Circuit Diagram of the Operation Setting Selection Switches



Caution The sample program cannot be used for the 78K0R/KC3-L microcontroller, because the 78K0R/KC3-L microcontroller does not have P14 to P17.

4.2.2 Settings to specify by using the operation setting selection switches

The items to set up by using the switches are described below.

(1) Operation mode

SW9 and SW8 are used to specify the operation mode.

SW9 (P51)	SW8 (P50)	Operation Mode
Low	Low	Intermittent STOP mode
Low	High	Intermittent HALT mode
High	Low	Real-time counter mode
High	High	Setting prohibited

(2) Clock used as the CPU/peripheral hardware clock

SW7 and SW6 are used to specify the clock used as the CPU/peripheral hardware clock.

SW7 (P17)	SW6 (P16)	CPU/Peripheral Hardware Clock
Low	Low	High-speed system clock
Low	High	Internal high-speed oscillation clock
High	Low	Subsystem clock
High	High	Setting prohibited

(3) Oscillation frequency

SW5 and SW4 are used to specify the oscillation frequency.

SW5 (P15)	SW4 (P14)	Oscillation Frequency
Low	Low	2 MHz or 1 MHz (internal high-speed oscillation clock)
Low	High	10 MHz or 8 MHz (internal high-speed oscillation clock)
High	Low	20 MHz
High	High	32.768 kHz (subsystem clock)

(4) Standby mode

SW3 and SW2 are used to specify the standby mode. For the sample program, the standby mode that can be used differs depending on the operation mode. Be sure to specify the correct standby mode.

SW3 (P13)	SW2 (P12)	Standby Mode
Low	Low	Normal operation mode
Low	High	HALT mode
High	Low	Sub-HALT mode
High	High	STOP mode

(5) Supply voltage (V_{DD})

SW1 and SW0 are used to specify the supply voltage. For the sample program, the oscillation frequency at which the sample program can be used and the A/D conversion speed differ depending on the supply voltage (V_{DD}). Be sure to specify the correct supply voltage (V_{DD}).

SW1 (P11)	SW0 (P10)	Supply Voltage (V_{DD})
Low	Low	5.0 [V]
Low	High	3.0 [V]
High	Low	1.8 [V]
High	High	Setting prohibited

Table 4-1 shows the operation modes and the corresponding switch settings.

Table 4-1. Operation Modes and the Corresponding Switch Settings

Item to Set Up	Switch Settings		Specified Mode and Value	
Operation mode	SW9	SW8		
	0	0	Intermittent STOP mode	<1>
	0	1	Intermittent HALT mode	<2>
	1	0	Real-time counter mode	<3>
	1	1	Setting prohibited	<4>
CPU/peripheral hardware clock	SW7	SW6		
	0	0	High-speed system clock	<5>
	0	1	Internal high-speed oscillation clock	<6>
	1	0	Subsystem clock	<7>
	1	1	Setting prohibited	<8>
Frequency	SW5	SW4		
	0	0	2 MHz or 1 MHz (internal high-speed oscillation clock)	<9>
	0	1	10 MHz or 8 MHz (internal high-speed oscillation clock)	<10>
	1	0	20 MHz (internal high-speed oscillation clock)	<11>
	1	1	32.768 kHz (subsystem clock)	<12>
Standby mode	SW3	SW2		
	0	0	Normal operation mode	<13>
	0	1	HALT mode	<14>
	1	0	Sub-HALT mode	<15>
	1	1	STOP mode	<16>
Operation voltage	SW1	SW0		
	0	0	5.0 [V]	<17>
	0	1	3.0 [V]	<18>
	1	0	1.8 [V]	<19>
	1	1	Setting prohibited	<20>

Caution The sample program cannot be used by using the following combinations of settings below. If any such combination is specified, respecify valid values, and then perform a reset.

<1> + <7>, <1> + <11>, <2> + <7>, <2> + <15>, <2> + <16>, <3> + <13>, <3> + <14>, <10> + <19>, <11> + <19>

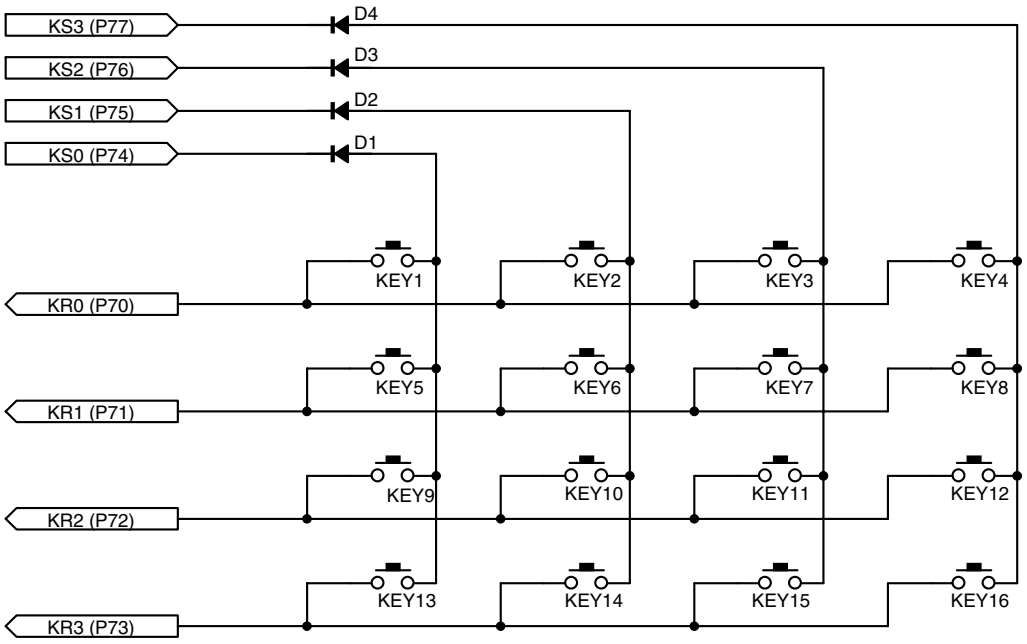
4.3 Key Matrix

The configuration of the key matrix to use is shown below. Figure 4-3 shows a circuit diagram of the key matrix.

Key Send / Key Return	KS0 (P74) ↓	KS1 (P75) ↓	KS2 (P76) ↓	KS3 (P77) ↓
KR0 (P70) ←	'KEY1'	'KEY2'	'KEY3'	'KEY4'
KR1 (P71) ←	'KEY5'	'KEY6'	'KEY7'	'KEY8'
KR2 (P72) ←	'KEY9'	'KEY10'	'KEY11'	'KEY12'
KR3 (P73) ←	'KEY13'	'KEY14'	'KEY15'	'KEY16'

In the sample program, only whether there are key entries is identified, and the same processing is executed regardless of the entered key. To specify processing to perform according to the entered key, add the required processing to the source.

Figure 4-3. Key Matrix Circuit Diagram



Caution The sample program cannot be used for the 78K0R/KC3-L microcontroller, because the 78K0R/KC3-L microcontroller does not have P76 and P77.

CHAPTER 5 SETUP

This chapter describes how to set up the option bytes and power reduction and specify the vector table, stack pointer, and clock. The processing is described only for the assembly source code, because it does not differ from the C source code.

For details about how to set up the registers, see the **78K0R/Kx3-L User's Manual (U19291E)**.

The highlighted sections in the register setup examples indicate the values specified in the sample program.

Example:

Operation control of watchdog timer counter

0	Counter operation stopped in HALT/STOP mode
1	Counter operation enabled in HALT/STOP mode

5.1 Setting Up the Option Bytes

The option bytes are set up in op.asm. The following settings are set up by using the option bytes:

(1) User option bytes

- Setting up the watchdog timer
- Setting up the LVI after a reset release (when the power is turned on)
- Specifying the frequency of the internal high-speed oscillation clock

(2) On-chip debugging user option bytes

Controlling on-chip debugging

Figure 5-1 shows the format of the user option bytes (000C0H/010C0H). In the sample program, the watchdog timer is specified not to be used.

Figure 5-1. Format of the User Option Bytes (000C0H/010C0H)

Address: 000C0H/010C0H[Note 1]

WDTINT	WINDOW1	WINDOW2	WDT0N	WDCS2	WDCS1	WDCS0	WDSTBYON

Operation control of watchdog timer counter

WDSTBYON	
0	Counter operation stopped in HALT/STOP mode[Note 2]
1	Counter operation enabled in HALT/STOP mode

Watchdog timer overflow time

WDCS2	WDCS1	WDCS0	
0	0	0	$2^{10}/f_{IL}$ (3.88 ms)
0	0	1	$2^{11}/f_{IL}$ (7.76 ms)
0	1	0	$2^{12}/f_{IL}$ (15.52 ms)
0	1	1	$2^{13}/f_{IL}$ (31.03 ms)
1	0	0	$2^{15}/f_{IL}$ (124.12 ms)
1	0	1	$2^{17}/f_{IL}$ (496.48 ms)
1	1	0	$2^{18}/f_{IL}$ (992.97 ms)
1	**1**	**1**	$2^{20}/f_{IL}$ (3971.88 ms)

Operation control of watchdog timer counter

WDT0N	
0	Counter operation stopped (counting stopped after reset release)
1	Counter operation enabled (counting started after reset release)

Watchdog timer window open period[Note 2]

WINDOW2	WINDOW1	
0	0	25%
0	1	50%
1	0	75%
1	**1**	100%

Use of interval interrupt of watchdog timer

WDTINT	
0	Interval interrupt is not used.
1	Interval interrupt is generated when 75% of the overflow time is reached.

Notes 1. Specify the same value as 000C0H to 010C0H when the boot swap operation is used because 000C0H is replaced by 010C0H.

2. The window open period is 100% when WDSTBYON = 0, regardless of the values of WINDOW1 and WINDOW0.

Figure 5-2 shows the format of the user option bytes (000C1H/010C1H). In the sample program, low-voltage detection is disabled when the power is turned on. Specify the setting for 1 MHz or the setting for 8 MHz or 20 MHz as the oscillation frequency of the internal high-speed oscillation clock.

Figure 5-2. Format of the User Option Bytes (000C1H/010C1H)

Address: 000C1H/010C1H[Note 1]

1	1	1	1	1	FRQSEL2	FRQSEL1	LVIOFF

Setting of LVI on power application

0	LVI is ON by default (LVI default start function enabled) upon reset release (upon power application)
1	LVI is OFF by default (LVI default start function stopped) upon reset release (upon power application)

Internal high-speed oscillation clock frequency

0	1	8 MHz or 20 MHz[Note 2] (default)
1	0	1 MHz[Note 3]
Other than the above		Setting prohibited

Notes 1. Specify the same value as 000C1H to 010C1H when the boot swap operation is used because 000C1H is replaced by 010C1H.

2. If 8 MHz or 20 MHz is selected, the 8 MHz internal high-speed oscillation clock automatically starts oscillating after reset release. To use the 20 MHz internal high-speed oscillation clock to operate the microcontroller, set up the 20 MHz internal high-speed oscillation control register.
The setting cannot be changed to the 1 MHz internal high-speed oscillation clock while the microcontroller operates.

3. If 1 MHz is selected, the microcontroller operates on the 1 MHz internal high-speed oscillation clock after reset release.
The setting cannot be changed to the 8 MHz or 20 MHz internal high-speed oscillation clock while the microcontroller operates.

Figure 5-3 shows the format of the user option bytes (000C2H/010C2H).

Figure 5-3. Format of the User Option Bytes (000C2H/010C2H)

Address: 000C2H/010C2H[Note]

1	1	1	1	1	1	1	1

Note Be sure to set FFH to 000C2H, because 000C2H is a reserved area. Also set FFH to 010C2H when the boot swap operation is used because 000C2H is replaced by 010C2H.

Figure 5-4 shows the format of the on-chip debug option bytes (000C3H/010C3H). Specify different values depending on how on-chip debugging is used.

Figure 5-4. Format of the On-Chip Debug Option Bytes (000C3H/010C3H)

Address: 000C3H/010C3H[Note]

OCDE NSET	0	0	0	0	1	0	OCDE RDS

Control of on-chip debug operation

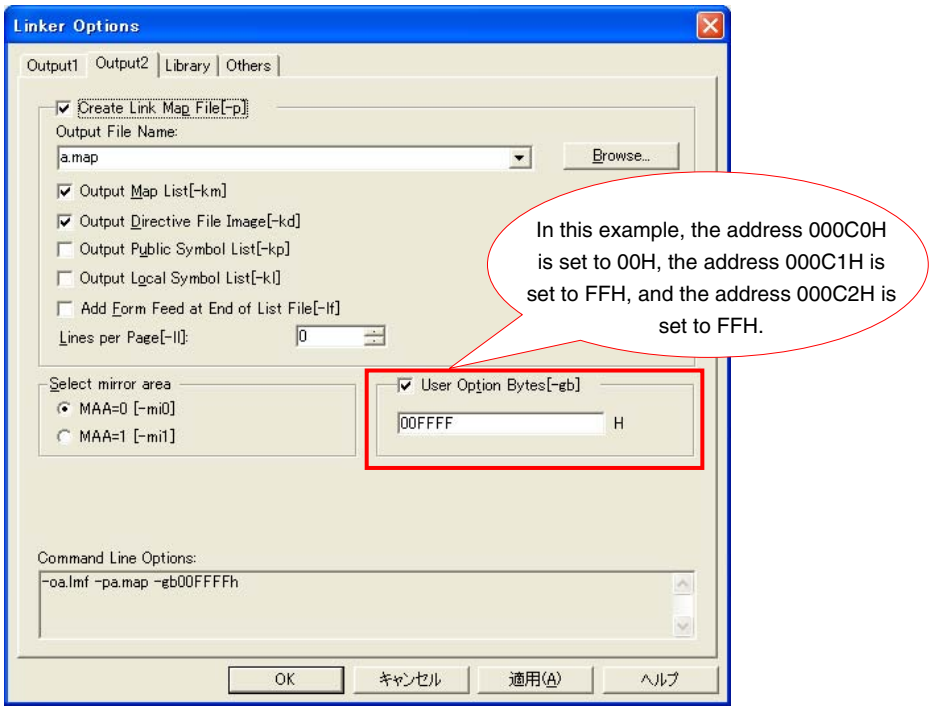
0	0	Disables on-chip debug operation.
0	1	Setting prohibited
1	0	Erases data of flash memory in case of failures in enabling on-chip debugging and authenticating on-chip debug security ID.
1	1	Does not erase data of flash memory in case of failures in enabling on-chip debugging and authenticating on-chip debug security ID.

Note Specify the same value as 000C3H to 010C3H when the boot swap operation is used because 000C3H is replaced by 010C3H.

The option bytes can be set up by using the linker option.

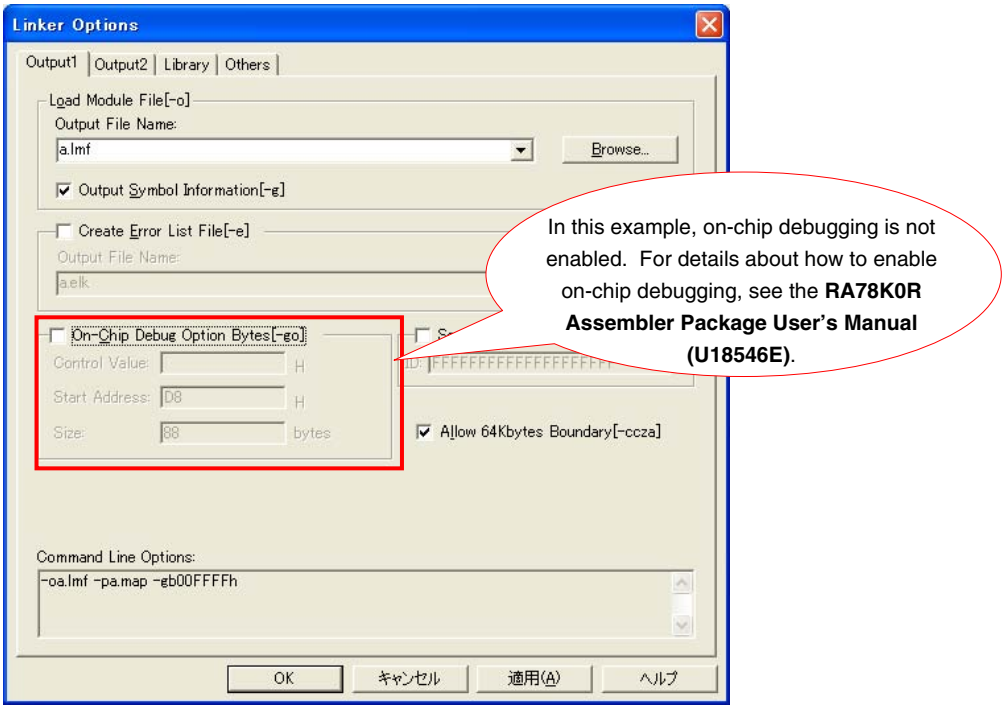
(1) Setting up the user option bytes

Select [Tool], [Linker Options], the [Output2] tab, and then the [User Option Bytes] check box in the PM+. Next, specify a value.



(2) Setting up the on-chip debug option bytes

Select [Tool], [Linker Options], the [Output1] tab, and then the [On-Chip Debug Option Bytes] check box in the PM+. Next, specify values.

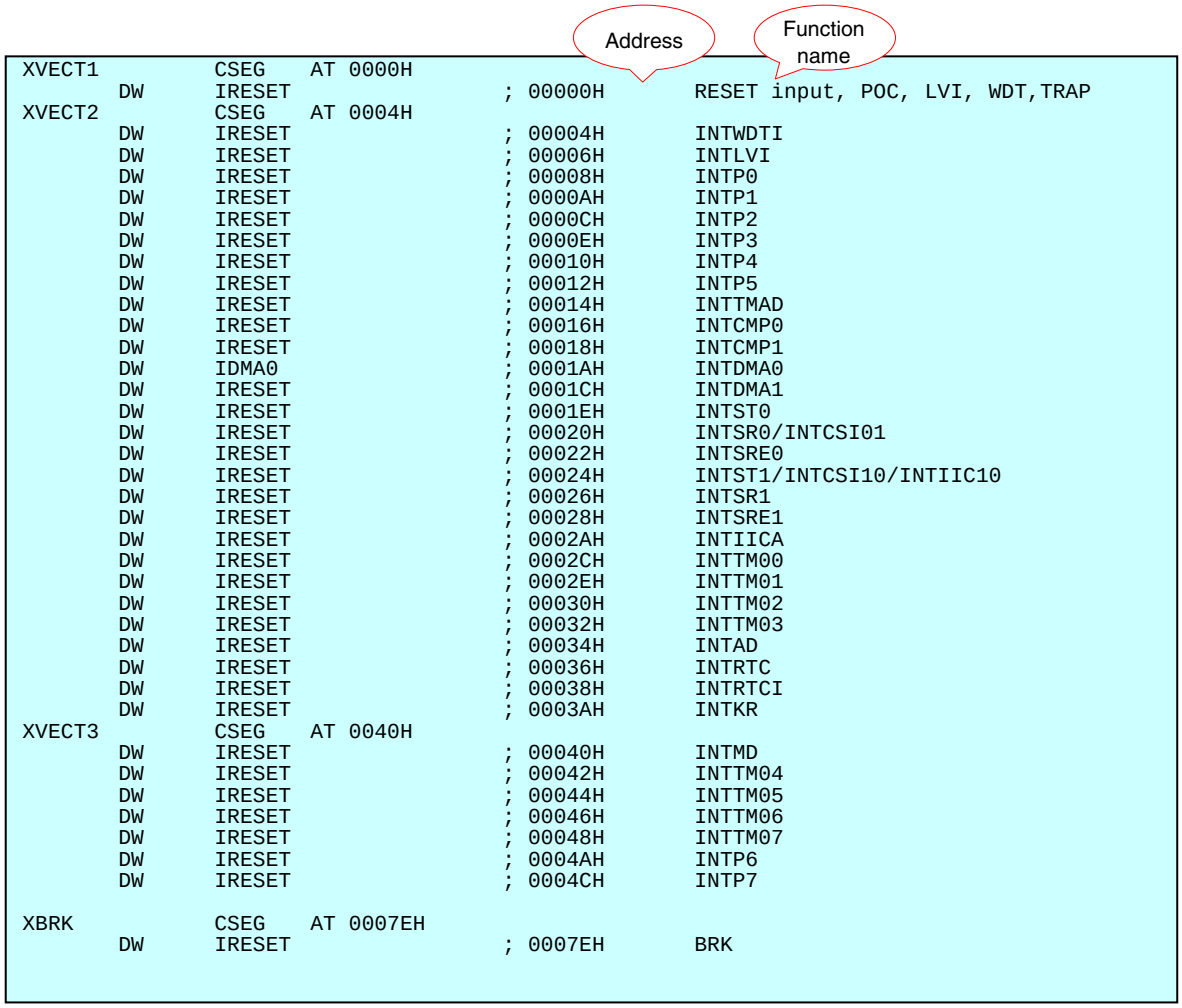


5.2 Setting Up the Vector Table

The vector table area stores the program start address when processing branches due to the generation of a reset or interrupt request.

In the sample program, only DMA0 transfer completion interrupt servicing and the reset vector used when starting a reset are set up. These settings are required when writing code in assembly language. A reset vector must be specified by using a #pragma directive when writing code in C. For details, see the **CC78K0R Language User's Manual (U18548E)**.

Example: Setting up only the reset vector used to start a reset (same settings as in the sample program)




```
XVECT1          CSEG    AT 0000H
        DW      IRESET                  ; 00000H        RESET input, POC, LVI, WDT,TRAP
XVECT2          CSEG    AT 0004H
        DW      IRESET                  ; 00004H        INTWDTI
        DW      IRESET                  ; 00006H        INTLVI
        DW      IRESET                  ; 00008H        INTP0
        DW      IRESET                  ; 0000AH        INTP1
        DW      IRESET                  ; 0000CH        INTP2
        DW      IRESET                  ; 0000EH        INTP3
        DW      IRESET                  ; 00010H        INTP4
        DW      IRESET                  ; 00012H        INTP5
        DW      IRESET                  ; 00014H        INTTMAD
        DW      IRESET                  ; 00016H        INTCMP0
        DW      IRESET                  ; 00018H        INTCMP1
        DW      IDMA0                   ; 0001AH        INTDMA0
        DW      IRESET                  ; 0001CH        INTDMA1
        DW      IRESET                  ; 0001EH        INTST0
        DW      IRESET                  ; 00020H        INTSR0/INTCSI01
        DW      IRESET                  ; 00022H        INTSRE0
        DW      IRESET                  ; 00024H        INTST1/INTCSI10/INTIIC10
        DW      IRESET                  ; 00026H        INTSR1
        DW      IRESET                  ; 00028H        INTSRE1
        DW      IRESET                  ; 0002AH        INTIICA
        DW      IRESET                  ; 0002CH        INTTM00
        DW      IRESET                  ; 0002EH        INTTM01
        DW      IRESET                  ; 00030H        INTTM02
        DW      IRESET                  ; 00032H        INTTM03
        DW      IRESET                  ; 00034H        INTAD
        DW      IRESET                  ; 00036H        INTRTC
        DW      IRESET                  ; 00038H        INTRTCI
        DW      IRESET                  ; 0003AH        INTKR
XVECT3          CSEG    AT 0040H
        DW      IRESET                  ; 00040H        INTMD
        DW      IRESET                  ; 00042H        INTTM04
        DW      IRESET                  ; 00044H        INTTM05
        DW      IRESET                  ; 00046H        INTTM06
        DW      IRESET                  ; 00048H        INTTM07
        DW      IRESET                  ; 0004AH        INTP6
        DW      IRESET                  ; 0004CH        INTP7

XBRK            CSEG    AT 0007EH
        DW      IRESET                  ; 0007EH        BRK
```

After a reset release, the program starts from the address specified using the reset vector (IRESET above).

In the sample program, vector table addresses other than 00000H and 0001AH are not used. IRESET is set to all other vector table addresses as for 00000H. By specifying these settings, the processing branches to IRESET and the same processing as that performed after a reset release is performed, even if an unintended interrupt occurs.

5.3 Specifying a Stack Pointer

The stack area temporarily stores data such as for the program counter, register values, and PSW (program status word). Only the internal high-speed RAM area can be specified as the stack area. The starting address of the stack area is specified by using a stack pointer and the stack area is allocated.

The stack area is used to execute the next instruction or when an interrupt occurs.

- PUSH, CALL, CALLT, BRK, or an interrupt: Data is saved to the stack area.
- POP, RET, RETI, RBRK: Data is restored from the stack area.

The stack area must be allocated when writing code in assembly language. The stack area does not have to be allocated when writing code in C, because this is automatically performed by the startup routine.

Example 1: Using FFEC0H to FFEDFH of the RAM area (32 bytes) as the stack area

```
DSTACK          DSEG    AT 0FFEC0H
RSTACK:         DS      20H
CSP:
          ⋮

XMAIN   CSEG    UNIT
IRESET:
          ⋮
        MOVW    SP,     #LOWW CSP
```

The stack area is allocated in the RAM area.

A stack pointer is specified after a reset release.

In this example, the address FFEE0H (= FFEC0H + 20H) is specified for the stack pointer.
By using the above code, the last 32 bytes of the RAM area (FFEC0H to FFEDFH) can be allocated as the stack area.

Example 2: Using FFE00H to FFE1FH of the RAM area (32 bytes) as the stack area
(The settings are the same as in the sample program)

```
DSTACK          DSEG    AT 0FFE00H
RSTACK:         DS      20H
CSP:
          ⋮

XMAIN   CSEG    UNIT
IRESET:
          ⋮
        MOVW    SP,     #LOWW CSP
```

The stack area is allocated in the RAM area.

A stack pointer is specified after a reset release.

In this example, the address FFE20H (= FFE00H + 20H) is specified for the stack pointer. By using these settings, the stack area can be allocated while avoiding the saddr area.

5.4 Specifying the Clock

(1) Specifying the clock operation mode

Both the operation mode of the X1/P121, X2/EXCLK/P122, XT1/P123, and XT2/P124 pins and the gain of the oscillator are selected by using the clock operation mode control register (CMC).

The operation mode of the pins and the gain of the oscillator are selected according to the clock used in the program. In the sample program, the high-speed system clock pin is set to X1 oscillation mode and the subsystem clock pin is set to XT1 oscillation mode, because the high-speed system clock and subsystem clock are used. The power consumed by the subsystem clock is reduced by setting the subsystem clock to ultra-low power consumption oscillation mode.

To use a high-speed system clock that is faster than 10 MHz, AMPH must be set to 1. In the sample program, the clock operation mode control register (CMC) is set up according to the operation setting parameters.

Figure 5-5. Format of the Clock Operation Mode Control Register (CMC)

Address: FFFA0H

EXCLK	OSCSEL	0	OSCSELS	0	AMPHS1	AMPHS0	AMPM

AMPH	Control of high-speed system clock oscillation frequency
0	2 MHz ≤ f_{MX} ≤ 10 MHz
1	10 MHz < f_{MX} ≤ 20 MHz

AMPHS1	AMPHS0	XT1 oscillator oscillation mode selection
0	0	Low power consumption oscillation (initial value after a reset release)
0	1	Normal oscillation
1	x	Ultra-low power consumption oscillation

OSCSELS	Subsystem clock pin operation mode	XT1/P123 pin	XT2/P124 pin
0	Input port mode	Input port	
1	XT1 oscillation mode	Crystal resonator connection	

EXCLK	OSCSEL	High-speed system clock pin operation mode	X1/P121 pin	X2/EXCLK/P122 pin
0	0	Input port mode	Input port	
0	1	X1 oscillation mode	Crystal/ceramic resonator connection	
1	0	Input port mode	Input port	
1	1	External clock input mode	Input port	External clock input

Cautions 1. Be sure to clear bits 5 and 3 to "0".

2. Note the following because the XT1 oscillator is a circuit with low amplification in order to achieve low power consumption:

- **Pins and circuit boards include parasitic capacitance. Therefore, evaluate oscillation based on the circuit board that will actually be used to confirm that there are no problems.**
- **When using the ultra-low power consumption oscillation (AMPHS1 = 1) as the mode of the XT1 oscillator, use the recommended resonators described in the electrical specifications in the 78K0R/Kx3-L User's Manual (U19291E).**
- **Make the wiring between the XT1 and XT2 pins and the resonators as short as possible, and minimize the parasitic capacitance and wiring resistance, especially when ultra-low power consumption oscillation (AMPHS1 = 1) is selected.**
- **Create the circuit board by using material that has little wiring resistance and parasitic capacitance.**
- **Place a ground pattern that has the same potential as V_{SS} as much as possible near the XT1 oscillator.**
- **Be sure that the signal lines between the XT1 and XT2 pins, and the resonators do not cross with the other signal lines. Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **The impedance between the XT1 and XT2 pins might drop and oscillation might be disturbed due to moisture absorption of the circuit board in a high-humidity environment or dew condensation on the board. When using the circuit board in such an environment, take measures to damp-proof the circuit board, such as by coating.**
- **When coating the circuit board, use material that does not cause capacitance or leakage between the XT1 and XT2 pins.**

Remarks 1. f_{MX}: High-speed system clock frequency

2. x: don't care

(2) Specifying the clock operation status

The operation of the high-speed system clock, internal high-speed oscillation clock, and subsystem clock are controlled by using the clock operation status control register (CSC).

In the sample program, the high-speed system clock, internal high-speed oscillation clock, and subsystem clock are set up according to the operation setting parameters.

Figure 5-6. Format of the Clock Operation Status Control Register (CSC)

Address: FFFA1H

MSTOP	XTSTOP	0	0	0	0	0	HIOSTOP

HIOSTOP	Internal high-speed oscillation clock operation control
0	Internal high-speed oscillation clock operating
1	Internal high-speed oscillation clock stopped

XTSTOP	Subsystem clock operation control	
	XT1 oscillation mode	Input port mode
0	Subsystem clock operating	–
1	Subsystem clock stopped	

MSTOP	High-speed system clock operation control		
	X1 oscillation mode	External clock input mode	Input port mode
0	High-speed system clock operating	External clock from EXCLK pin is valid	–
1	High-speed system clock stopped	External clock from EXCLK pin is invalid	

Caution Be sure to clear bits 5 to 1 to “0”.

(3) Selecting the oscillation stabilization time

The oscillation stabilization time for the high-speed system clock after the STOP mode is exited is selected.

If the high-speed system clock is selected as the CPU clock, the system automatically waits for the time specified by using the oscillation stabilization time select register (OSTS) after the STOP mode is exited.

If the internal high-speed oscillation clock is selected as the CPU clock, check whether the oscillation stabilization time has elapsed after the STOP mode is exited by using the oscillation stabilization time counter status register (OSTC). By using the oscillation stabilization time counter status register, the oscillation stabilization time can be checked up to the time specified using the oscillation stabilization time select register. Specify values according to the resonator to use.

Figure 5-7. Format of the Oscillation Stabilization Time Select Register (OSTS)

Address: FFFA3H

0	0	0	0	0	OSTS2	OSTS1	OSTS0

OSTS2	OSTS1	OSTS0	Oscillation stabilization time selection
0	0	0	$2^8/f_X$
0	0	1	$2^9/f_X$
0	1	0	$2^{10}/f_X$
0	1	1	$2^{11}/f_X$
1	0	0	$2^{13}/f_X$
1	0	1	$2^{15}/f_X$
1	1	0	$2^{17}/f_X$
1	1	1	$2^{18}/f_X$

Caution Be sure to clear bits 7 to 3 to “0”.

(4) Specifying the operation speed mode

The boost circuit for using flash memory at high speeds is controlled by using bits 1 and 0 (FLPC and FSEL) of the operation speed mode control register (OSMC).

When using the system clock at a low speed of 10 MHz or less, power consumption can be reduced by setting FLPC and FSEL to 0.

Bit 7 (RTCLPC) can be used to control whether the subsystem clock signal is supplied to peripheral functions in sub-HALT mode. Current consumption in sub-HALT mode can be reduced by setting RTCLPC to 1. However, the clock signal can no longer be supplied to peripheral functions other than the real-time counter.

[Conditions for reducing current consumption in sub-HALT mode]

- RTCLPC is set to 1.
- Bits other than the real-time counter of a peripheral enable register (PERn) are 0.

Figure 5-8 shows the format of the operation speed mode control register.

Figure 5-8. Format of the Operation Speed Mode Control Register (OSMC)

Address: F00F3H

RTCLPC	**0**	**0**	**0**	**0**	**0**	FLPC	FSEL

FLPC	FSEL	f_{CLK} frequency selection
0	0	Operates at a frequency of 10 MHz or less (default).
0	1	Operates at a frequency higher than 10 MHz.
1	0	Operates at a frequency of 1 MHz.
1	1	Setting prohibited

RTCLPC	Setting in STOP mode or sub-HALT mode
0	Enables supply of subsystem clock to peripheral functions
1	Stops supply of subsystem clock to peripheral functions other than real-time counter

Cautions 1. Be sure to clear bits 6 to 2 to "0".

2. The operation speed mode control register can be written only once after a reset.

(5) Selecting the system clock

The clock used as the CPU/peripheral hardware clock is selected and the division ratio is specified.

In the sample program, the system clock control register (CKC) is set up according to the operation setting parameters.

Figure 5-9. Format of the System Clock Control Register (CKC)

Address: FFFA4H

CLS	CSS	MCS	MCM0	1	MDIV2	MDIV1	MDIV0

CSS	MCM0	MDIV2	MDIV1	MDIV0	Selection of CPU/peripheral hardware clock (f_{CLK})
0	0	0	0	0	f_{IH}
		0	0	1	f_{IH}/2 (default)
		0	1	0	$f_{IH}/2^2$
		0	1	1	$f_{IH}/2^3$
		1	0	0	$f_{IH}/2^4$
		1	0	1	$f_{IH}/2^5$
0	1	0	0	0	f_{MX}
		0	0	1	f_{MX}/2
		0	1	0	$f_{MX}/2^2$
		0	1	1	$f_{MX}/2^3$
		1	0	0	$f_{MX}/2^4$
		1	0	1	$f_{MX}/2^5$ (Setting prohibited if $f_{MX} < 4$ MHz)
1	x	x	x	x	f_{SUB}/2
Other than the above					Setting prohibited

MCS	Status of main system clock (f_{MAIN})
0	Internal high-speed oscillation clock (f_{IH})
1	High-speed system clock (f_{MX})

CLS	Status of CPU/peripheral hardware clock (f_{CLK})
0	Main system clock (f_{MAIN})
1	Subsystem clock (f_{SUB})

Cautions 1. Be sure to set bit 3 to "1".

2. Bits 7 and 5 are read-only.

Remarks 1. f_{CLK}: CPU/peripheral hardware clock frequency
f_{MX}: High-speed system clock frequency
f_{IH}: Internal high-speed oscillation clock frequency
f_{SUB}: Subsystem clock frequency

2. x: don't care

(6) Controlling the 20 MHz internal high-speed oscillation clock

The 20 MHz internal high-speed oscillation clock is controlled by using the 20 MHz internal high-speed oscillation control register (DSCCTL).

This register can be used to select whether to use the 20 MHz internal high-speed oscillation clock (f_{IH20}) as a peripheral hardware clock that supports 20 MHz operation.

DSCCTL can be set up by using a 1-bit or 8-bit memory manipulation instruction.

The generation of a reset signal sets this register to 00H.

Figure 5-10. Format of the 20 MHz Internal High-Speed Oscillation Control Register (DSCCTL)

Address: F00F6H

0	0	0	0	DSCS	SELDSC	0	DSCON

DSCON	20 MHz internal high-speed oscillation clock (f_{IH20}) operation enable/disable
0	Disables operation.
1	Enables operation.

SELDSC	Selection of 20 MHz internal high-speed oscillation for CPU/peripheral hardware clock (f_{CLK})
0	Does not select 20 MHz internal high-speed oscillation (clock selected by CKC register is supplied to f_{CLK})
1	Selects 20 MHz internal high-speed oscillation (20 MHz internal high-speed oscillation is supplied to f_{CLK})

Cautions 1. Be sure to clear bits 7 to 4 and 1 to "0".

2. Bit 3 is read-only.

(7) Selecting peripheral enable register 0, 1, or 2 (PER0, PER1, or PER2)

Whether to use each peripheral function is specified by using peripheral enable register n (PERn: n = 0 to 2). Power consumption and noise can be reduced by stopping the supply of the clock signal to unused hardware. In the sample program, only the peripheral functions to use are enabled according to the operation mode.

Figure 5-11. Format of Peripheral Enable Register 0 (PER0)

Address: F00F0H

RTCEN	0	ADCEN	IICAEN	0	SAU0EN	0	0

SAU0EN	Control of serial array unit input clock supply
0	Stops input clock supply. • SFR used by the serial array unit cannot be written. • The serial array unit is in the reset status.
1	Supplies input clock. • SFR used by the serial array unit can be read and written.

IICAEN	Control of serial interface IICA input clock supply
0	Stops input clock supply. • SFR used by the serial interface IICA cannot be written. • The serial interface IICA is in the reset status.
1	Supplies input clock. • SFR used by the serial interface IICA can be read and written.

ADCEN	Control of A/D converter input clock supply
0	Stops input clock supply. • SFR used by the A/D converter cannot be written. • The A/D converter is in the reset status.
1	Supplies input clock. • SFR used by the A/D converter can be read and written.

RTCEN	Control of real-time counter (RTC) input clock supply[Note]
0	Stops input clock supply. • SFR used by the real-time counter (RTC) cannot be written (can be read). • Operation of the real-time counter (RTC) continues.
1	Supplies input clock. • SFR used by the real-time counter (RTC) can be read and written.

Note The input clock that can be controlled by using RTCEN is used when the register used by the real-time counter (RTC) is accessed from the CPU. RTCEN cannot control supply of the operation clock (f_{SUB}) of the real-time counter.

Caution Be sure to clear bits 6, 3, 1, and 0 to "0".

Figure 5-12. Format of Peripheral Enable Register 1 (PER1)

Address: F00F1H

0	0	0	0	OACMPEN	0	0	0

OACMPEN	Control of programmable gain amplifier input clock supply
0	Stops input clock supply. • SFR used by the programmable gain amplifier cannot be written. • The programmable gain amplifier is in the reset status.
1	Supplies input clock. • SFR used by the programmable gain amplifier can be read and written.

Caution Be sure to clear bits 7 to 4 and 2 to 0 to "0".

Figure 5-13. Format of Peripheral Enable Register 2 (PER2)

Address: F00F2H

0	0	0	0	0	0	0	TAU0EN

TAU0EN	Control of timer array unit TAUS input clock supply
0	Stops input clock supply. • SFR used by timer array unit TAUS cannot be written. • Timer array unit TAUS is in the reset status.
1	Supplies input clock. • SFR used by timer array unit TAUS can be read and written.

Caution Be sure to clear bits 7 to 1 to "0".

(8) Selecting the regulator output voltage

The regulator output voltage is selected by using the regulator mode control register (RMC). The power consumption by the 78K0R/Kx3-L can be reduced by decreasing the regulator output voltage. The RMC register can be used only under the following conditions when the regulator output voltage is fixed to low current consumption mode.

<When the X1 clock is selected as the CPU clock>

$f_X \leq 5$ MHz and $f_{CLK} \leq 1$ MHz

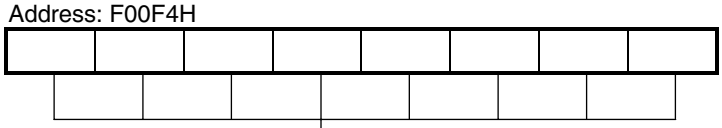
<When the internal high-speed oscillation clock, external input clock, or subsystem clock is selected as the CPU clock>

$f_{CLK} \leq 1$ MHz

Figure 5-14 shows the format of the regulator mode control register. The next page describes how to set up the register by using the source code.

Figure 5-14. Format of the Regulator Mode Control Register (RMC)

Address: F00F4H



RMC[7:0]	Control of regulator output voltage
5AH	Fixed to low current consumption mode (1.8 V)
00H	Switches normal current mode (2.4 V) and low current consumption mode (1.8 V) according to the condition
Other than the above	Setting prohibited

Cautions 1. Rewrite the RMC register while the CPU operates on the subsystem clock (f_{SUB}) and the high-speed system clock (f_{MX}), internal high-speed oscillation clock (f_{IH}), and 20 MHz internal high-speed oscillation clock (f_{IH20}) are stopped.

2. Self-programming cannot be used in low current consumption mode.

The following source code shows how to switch the regulator mode control register when the CPU is operating on the internal high-speed oscillation clock (f_{IH}), the high-speed system clock (f_{MX}) and 20 MHz internal high-speed oscillation clock (f_{IH20}) are stopped, and the subsystem clock (f_{SUB}) is oscillating.

```
       SET1    CSS                   ;CPU clock: XT1 oscillator
JCLK100:
       BF      CLS,    $JCLK100      ;Is the CPU clock operating on the XT1 oscillator? No,

       ;Stop oscillators other than the XT1 oscillator to set up the regulator.
       SET1    MSTOP                 ;Stops the X1 and X2 oscillators.
       SET1    HIOSTOP               ;Stops the internal high-speed oscillator.

       ;Specify the regulator low current consumption.
       MOV     RMC,    #5AH          ;Fixes the voltage to low current consumption mode
(1.8 V).

       ;Re-set the CPU clock to operate on the internal high-speed oscillator.
       CLR1    HIOSTOP               ;Starts the internal high-speed oscillator.
       NOP                           ;The system waits for the internal high-speed oscillator to
stabilize.

       CLR1    CSS                   ;Switches the CPU clock from the subclock to the
main clock.
       CLR1    MCM0                  ;CPU clock: Internal high-speed oscillator
JCLK200:
       BT      CLS,    $JCLK200      ;Is the CPU clock operating on the internal high-
speed oscillator? No,
```

CHAPTER 6 POWER MEASUREMENT RESULTS

This chapter describes the results of measuring the power consumption when using the sample program. Power consumption is measured only using the assembly language version, because the processing is the same as in the C version.

Note that all values are only examples and are not guaranteed to be the same.

(1) STOP mode processing

(a) Measurement results when V_{DD} = 1.8 [V]

Table 6-1. Current Consumption in Intermittent STOP Mode (SW9, SW8 = Low) When V_{DD} = 1.8 [V]

CPU/Peripheral Hardware Clock Frequency	Operation Mode	Peripheral Functions Operating in Standby Mode	Power Consumption	
			Standby Mode	Normal Operation Mode
X1/X2 oscillator (2 MHz)	Normal operation mode	*1	–	0.757 [mA]
	HALT mode	*2	0.15 [mA]	–
	STOP mode	*3	0.33 [μA]	–
Internal high-speed oscillator (1 MHz)	Normal operation mode	*4	–	0.207 [mA]
	HALT mode	*5	47.0 [μA]	–
	STOP mode	*3	0.33 [μA]	–
XT1 oscillator (32.768 kHz)	Normal operation mode	*6	–	3.95 [μA]
	Sub-HALT mode	*7	1.23 [μA]	–

*1: The X1 oscillator, XT1 oscillator, timer array unit, and key interrupt function operate.
*2: The X1 oscillator, XT1 oscillator, and key interrupt function operate.
*3: The key interrupt function operates.
*4: The internal high-speed oscillator, XT1 oscillator, timer array unit, and key interrupt function operate.
*5: The internal high-speed oscillator, XT1 oscillator, and key interrupt function operate.
*6: The XT1 oscillator, timer array unit, and key interrupt function operate.
*7: The XT1 oscillator and key interrupt function operate.

(b) Measurement results when V_{DD} = 3.0 [V]

Table 6-2. Current Consumption in Intermittent STOP Mode (SW9, SW8 = Low) When V_{DD} = 3.0 [V]

CPU/Peripheral Hardware Clock Frequency	Operation Mode	Peripheral Functions Operating in Standby Mode	Power Consumption	
			Standby Mode	Normal Operation Mode
X1/X2 oscillator (2 MHz)	Normal operation mode	*1	–	0.967 [mA]
	HALT mode	*2	0.250 [mA]	–
	STOP mode	*3	0.33 [μA]	–
X1/X2 oscillator (10 MHz)	Normal operation mode	*1	–	3.636 [μA]
	HALT mode	*2	0.596 [mA]	–
	STOP mode	*3	0.33 [μA]	–
X1/X2 oscillator (20 MHz)	Normal operation mode	*1	–	6.443 [mA]
	HALT mode	*2	1.187 [mA]	–
	STOP mode	*3	0.33 [μA]	–
Internal high-speed oscillator (1 MHz)	Normal operation mode	*4	–	0.209 [mA]
	HALT mode	*5	47.4 [μA]	–
	STOP mode	*3	0.33 [μA]	–
Internal high-speed oscillator (8 MHz)	Normal operation mode	*4	–	2.900 [mA]
	HALT mode	*5	0.413 [mA]	–
	STOP mode	*3	0.33 [μA]	–
Internal high-speed oscillator (20 MHz)	Normal operation mode	*6	–	6.482 [mA]
	HALT mode	*5	1.206 [mA]	–
	STOP mode	*3	0.33 [μA]	–
XT1 oscillator (32.768 kHz)	Normal operation mode	*7	–	4.00 [μA]
	Sub-HALT mode	*8	1.25 [μA]	–

*1: The X1 oscillator, XT1 oscillator, timer array unit, and key interrupt function operate.
*2: The X1 oscillator, XT1 oscillator, and key interrupt function operate.
*3: The key interrupt function operates.
*4: The internal high-speed oscillator, XT1 oscillator, timer array unit, and key interrupt function operate.
*5: The internal high-speed oscillator, XT1 oscillator, and key interrupt function operate.
*6: The 20 MHz internal high-speed oscillator, XT1 oscillator, timer array unit, and key interrupt function operate.
*7: The XT1 oscillator, timer array unit, and key interrupt function operate.
*8: The XT1 oscillator and key interrupt function operate.

(c) Measurement results when V_{DD} = 5.0 [V]

Table 6-3. Current Consumption in Intermittent STOP Mode (SW9, SW8 = Low) When V_{DD} = 5.0 [V]

CPU/Peripheral Hardware Clock Frequency	Operation Mode	Peripheral Functions Operating in Standby Mode	Power Consumption	
			Standby Mode	Normal Operation Mode
X1/X2 oscillator (2 MHz)	Normal operation mode	*1	–	0.975 [mA]
	HALT mode	*2	0.255 [mA]	–
	STOP mode	*3	0.33 [μA]	–
X1/X2 oscillator (10 MHz)	Normal operation mode	*1	–	3.653 [mA]
	HALT mode	*2	0.588 [mA]	–
	STOP mode	*3	0.33 [μA]	–
X1/X2 oscillator (20 MHz)	Normal operation mode	*1	–	6.471 [mA]
	HALT mode	*2	1.195 [mA]	–
	STOP mode	*3	0.33 [μA]	–
Internal high-speed oscillator (1 MHz)	Normal operation mode	*4	–	0.219 [mA]
	HALT mode	*5	47.6 [μA]	–
	STOP mode	*3	0.33 [μA]	–
Internal high-speed oscillator (8 MHz)	Normal operation mode	*4	–	2.913 [mA]
	HALT mode	*5	0.419 [mA]	–
	STOP mode	*3	0.33 [μA]	–
Internal high-speed oscillator (20 MHz)	Normal operation mode	*6	–	6.504 [mA]
	HALT mode	*5	1.212 [mA]	–
	STOP mode	*3	0.33 [μA]	–
XT1 oscillator (32.768 kHz)	Normal operation mode	*7	–	4.07 [μA]
	Sub-HALT mode	*8	1.30 [μA]	–

*1: The X1 oscillator, XT1 oscillator, timer array unit, and key interrupt function operate.
*2: The X1 oscillator, XT1 oscillator, and key interrupt function operate.
*3: The key interrupt function operates.
*4: The internal high-speed oscillator, XT1 oscillator, timer array unit, and key interrupt function operate.
*5: The internal high-speed oscillator, XT1 oscillator, and key interrupt function operate.
*6: The 20 MHz internal high-speed oscillator, XT1 oscillator, timer array unit, and key interrupt function operate.
*7: The XT1 oscillator, timer array unit, and key interrupt function operate.
*8: The XT1 oscillator and key interrupt function operate.

(2) HALT mode processing

(a) Measurement results when V_{DD} = 1.8 [V]

Table 6-4. Current Consumption in Intermittent HALT Mode (SW9 = Low, SW8 = High) When V_{DD} = 1.8 [V]

CPU/Peripheral Hardware Clock Frequency	Operation Mode	Peripheral Functions Operating in Standby Mode	Power Consumption
X1/X2 oscillator (2 MHz)	Normal operation mode	*1	0.691 [mA]
	HALT mode	*1	0.375 [mA]
Internal high-speed oscillator (1 MHz)	Normal operation mode	*2	0.338 [mA]
	HALT mode	*2	0.248 [mA]

*1: The X1 oscillator, XT1 oscillator, A/D converter, and DMA controller operate.

*2: The internal high-speed oscillator, XT1 oscillator, A/D converter, and DMA controller operate.

(b) Measurement results when V_{DD} = 3.0 [V]

Table 6-5. Current Consumption in Intermittent HALT Mode (SW9 = Low, SW8 = High) When V_{DD} = 3.0 [V]

CPU/Peripheral Hardware Clock Frequency	Operation Mode	Peripheral Functions Operating in Standby Mode	Power Consumption
X1/X2 oscillator (2 MHz)	Normal operation mode	*1	1.578 [mA]
	HALT mode	*1	1.100 [mA]
X1/X2 oscillator (10 MHz)	Normal operation mode	*1	3.906 [mA]
	HALT mode	*1	1.911 [mA]
X1/X2 oscillator (20 MHz)	Normal operation mode	*1	6.330 [mA]
	HALT mode	*1	2.954 [mA]
Internal high-speed oscillator (1 MHz)	Normal operation mode	*2	0.835 [mA]
	HALT mode	*2	0.744 [mA]
Internal high-speed oscillator (8 MHz)	Normal operation mode	*2	3.247 [mA]
	HALT mode	*2	1.618 [mA]
Internal high-speed oscillator (20 MHz)	Normal operation mode	*3	6.363 [mA]
	HALT mode	*3	2.976 [mA]

*1: The X1 oscillator, XT1 oscillator, A/D converter, and DMA controller operate.

*2: The internal high-speed oscillator, XT1 oscillator, A/D converter, and DMA controller operate.

*3: The 20 MHz internal high-speed oscillator, XT1 oscillator, A/D converter, and DMA controller operate.

(c) Measurement results when V_{DD} = 5.0 [V]

Table 6-6. Current Consumption in Intermittent HALT Mode (SW9 = Low, SW8 = High) When V_{DD} = 5.0 [V]

CPU/Peripheral Hardware Clock Frequency	Operation Mode	Peripheral Functions Operating in Standby Mode	Power Consumption
X1/X2 oscillator (2 MHz)	Normal operation mode	*1	2.542 [mA]
	HALT mode	*1	2.097 [mA]
X1/X2 oscillator (10 MHz)	Normal operation mode	*1	4.880 [mA]
	HALT mode	*1	3.028 [mA]
X1/X2 oscillator (20 MHz)	Normal operation mode	*1	7.292 [mA]
	HALT mode	*1	4.218 [mA]
Internal high-speed oscillator (1 MHz)	Normal operation mode	*2	1.850 [mA]
	HALT mode	*2	1.750 [mA]
Internal high-speed oscillator (8 MHz)	Normal operation mode	*2	4.224 [mA]
	HALT mode	*2	2.730 [mA]
Internal high-speed oscillator (20 MHz)	Normal operation mode	*3	7.322 [mA]
	HALT mode	*3	4.242 [mA]

*1: The X1 oscillator, XT1 oscillator, A/D converter, and DMA controller operate.

*2: The internal high-speed oscillator, XT1 oscillator, A/D converter, and DMA controller operate.

*3: The 20 MHz internal high-speed oscillator, XT1 oscillator, A/D converter, and DMA controller operate.

(3) Real-time counter mode

(a) Measurement results when V_{DD} = 1.8 [V]

Table 6-7. Current Consumption in Real-Time Counter Mode (SW9 = High, SW8 = Low) When V_{DD} = 1.8 [V]

CPU/Peripheral Hardware Clock Frequency	Operation Mode	Peripheral Functions Operating in Standby Mode	Power Consumption	
			Standby Mode	Normal Operation Mode
X1/X2 oscillator (2 MHz)	STOP mode	*1	0.91 [μA]	0.640 [mA]
Internal high-speed oscillator (1 MHz)	STOP mode	*1	0.91 [μA]	0.330 [mA]
XT1 oscillator (32.768 kHz)	Sub-HALT mode	*1	0.91 [μA]	4.340 [μA]

*1: The XT1 oscillator, real-time counter, and key interrupt function operate.

(b) Measurement results when V_{DD} = 3.0 [V]

Table 6-8. Current Consumption in Real-Time Counter Mode (SW9 = High, SW8 = Low) When V_{DD} = 3.0 [V]

CPU/Peripheral Hardware Clock Frequency	Operation Mode	Peripheral Functions Operating in Standby Mode	Power Consumption	
			Standby Mode	Normal Operation Mode
X1/X2 oscillator (2 MHz)	STOP mode	*1	0.91 [μA]	1.100 [mA]
X1/X2 oscillator (10 MHz)	STOP mode	*1	0.91 [μA]	3.825 [mA]
X1/X2 oscillator (20 MHz)	STOP mode	*1	0.91 [μA]	6.511 [mA]
Internal high-speed oscillator (1 MHz)	STOP mode	*1	0.91 [μA]	0.205 [mA]
Internal high-speed oscillator (8 MHz)	STOP mode	*1	0.91 [μA]	3.111 [mA]
Internal high-speed oscillator (20 MHz)	STOP mode	*1	0.91 [μA]	6.542 [mA]
XT1 oscillator (32.768 kHz)	Sub-HALT mode	*1	0.91 [μA]	4.300 [μA]

*1: The XT1 oscillator, real-time counter, and key interrupt function operate.

(c) Measurement results when V_{DD} = 5.0 [V]

Table 6-9. Current Consumption in Real-Time Counter Mode (SW9 = High, SW8 = Low) When V_{DD} = 5.0 [V]

CPU/Peripheral Hardware Clock Frequency	Operation Mode	Peripheral Functions Operating in Standby Mode	Power Consumption	
			Standby Mode	Normal Operation Mode
X1/X2 oscillator (2 MHz)	STOP mode	*1	0.91 [μA]	1.097 [mA]
X1/X2 oscillator (10 MHz)	STOP mode	*1	0.91 [μA]	3.846 [mA]
X1/X2 oscillator (20 MHz)	STOP mode	*1	0.91 [μA]	6.542 [mA]
Internal high-speed oscillator (1 MHz)	STOP mode	*1	0.91 [μA]	0.205 [mA]
Internal high-speed oscillator (8 MHz)	STOP mode	*1	0.91 [μA]	3.128 [mA]
Internal high-speed oscillator (20 MHz)	STOP mode	*1	0.91 [μA]	6.566 [mA]
XT1 oscillator (32.768 kHz)	Sub-HALT mode	*1	0.91 [μA]	4.440 [μA]

*1: The XT1 oscillator, real-time counter, and key interrupt function operate.

CHAPTER 7 RELATED DOCUMENTS

Document Name		Japanese/English
78K0R/Kx3-L User's Manual		PDF
78K0R Series Instructions User's Manual		PDF
RA78K0R Assembler Package User's Manual	Language	PDF
	Operation	PDF
CC78K0R C Compiler User's Manual	Language	PDF
	Operation	PDF
PM+ User's Manual		PDF

APPENDIX A REVISION HISTORY

Edition	Date Published	Page	Revision
1st edition	March 2009	–	–

For further information,
please contact:

NEC Electronics Corporation
1753, Shimonumabe, Nakahara-ku,
Kawasaki, Kanagawa 211-8668,
Japan
Tel: 044-435-5111
http://www.necel.com/

[America]

NEC Electronics America, Inc.
2880 Scott Blvd.
Santa Clara, CA 95050-2554, U.S.A.
Tel: 408-588-6000
800-366-9782
http://www.am.necel.com/

[Europe]

NEC Electronics (Europe) GmbH
Arcadiastrasse 10
40472 Düsseldorf, Germany
Tel: 0211-65030
http://www.eu.necel.com/

Hanover Office
Podbielskistrasse 166 B
30177 Hannover
Tel: 0 511 33 40 2-0

Munich Office
Werner-Eckert-Strasse 9
81829 München
Tel: 0 89 92 10 03-0

Stuttgart Office
Industriestrasse 3
70565 Stuttgart
Tel: 0 711 99 01 0-0

United Kingdom Branch
Cygnus House, Sunrise Parkway
Linford Wood, Milton Keynes
MK14 6NP, U.K.
Tel: 01908-691-133

Succursale Française
9, rue Paul Dautier, B.P. 52
78142 Velizy-Villacoublay Cédex
France
Tel: 01-3067-5800

Sucursal en España
Juan Esplandiu, 15
28007 Madrid, Spain
Tel: 091-504-2787

Tyskland Filial
Täby Centrum
Entrance S (7th floor)
18322 Täby, Sweden
Tel: 08 638 72 00

Filiale Italiana
Via Fabio Filzi, 25/A
20124 Milano, Italy
Tel: 02-667541

Branch The Netherlands
Steijgerweg 6
5616 HS Eindhoven
The Netherlands
Tel: 040 265 40 10

[Asia & Oceania]

NEC Electronics (China) Co., Ltd
7th Floor, Quantum Plaza, No. 27 ZhiChunLu Haidian
District, Beijing 100083, P.R.China
Tel: 010-8235-1155
http://www.cn.necel.com/

Shanghai Branch
Room 2509-2510, Bank of China Tower,
200 Yincheng Road Central,
Pudong New Area, Shanghai, P.R.China P.C:200120
Tel:021-5888-5400
http://www.cn.necel.com/

Shenzhen Branch
Unit 01, 39/F, Excellence Times Square Building,
No. 4068 Yi Tian Road, Futian District, Shenzhen,
P.R.China P.C:518048
Tel:0755-8282-9800
http://www.cn.necel.com/

NEC Electronics Hong Kong Ltd.
Unit 1601-1613, 16/F., Tower 2, Grand Century Place,
193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: 2886-9318
http://www.hk.necel.com/

NEC Electronics Taiwan Ltd.
7F, No. 363 Fu Shing North Road
Taipei, Taiwan, R. O. C.
Tel: 02-8175-9600
http://www.tw.necel.com/

NEC Electronics Singapore Pte. Ltd.
238A Thomson Road,
#12-08 Novena Square,
Singapore 307684
Tel: 6253-8311
http://www.sg.necel.com/

NEC Electronics Korea Ltd.
11F., Samik Lavied'or Bldg., 720-2,
Yeoksam-Dong, Kangnam-Ku,
Seoul, 135-080, Korea
Tel: 02-558-3737
http://www.kr.necel.com/

G0706