

WIN-63-0085-05

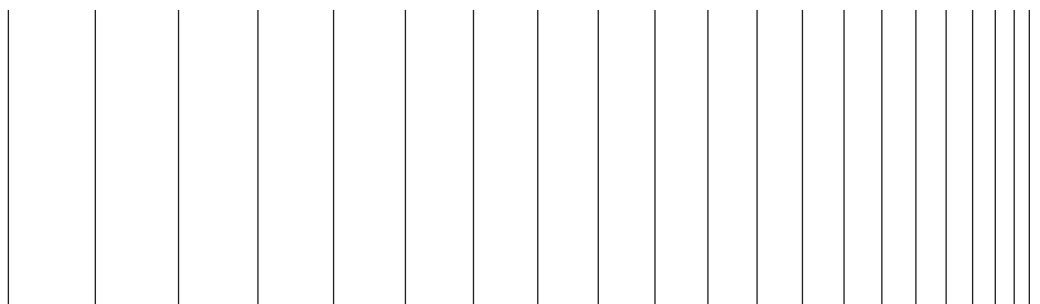
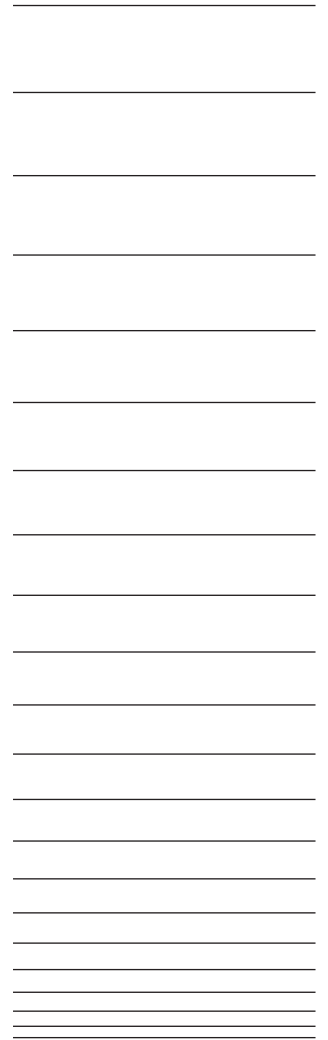


**HITACHI**

HITACHI FA COMPUTER

# **HF-W2000** Model 35/30

RAS FEATURES MANUAL



**USER'S  
MANUAL**



First Edition, February 2012, WIN-63-0085 (out of print)  
Second Edition, March 2012, WIN-63-0085-01 (out of print)  
Third Edition, June 2012, WIN-63-0085-02 (out of print)  
Fourth Edition, July 2012, WIN-63-0085-03 (out of print)  
Fifth Edition, June 2013, WIN-63-0085-04 (out of print)  
Sixth Edition, April 2015, WIN-63-0085-05

All Rights Reserved, Copyright © 2012, 2015, Hitachi Industry & Control Solutions, Ltd.

The contents of this publication may be revised without prior notice.

No part of this publication may be reproduced in any form or by any means without permission in writing from the publisher.

Printed in Japan.

TP<IC> (FL-MW2007)



## SAFETY SUMMARY

Before using this device, read the following safety precautions carefully to understand them.

- Follow the instructions and procedures in this manual during operation.
- Take particular care to adhere to safety-related precautions indicated on the device and in this manual. Failure to do so may result in personal harm or cause damage to the device.
- The safety-related precautions are described after the headers as shown below. They consist of a safety warning symbol and a header word of “DANGER”, “WARNING”, “CAUTION”, or “NOTICE”.



This is a safety warning symbol. It is used to call attention to potential dangers that could cause bodily harm. To avoid injury or death, follow the safety-related instruction that follows this symbol.



**DANGER:** Indicates the existence of dangers likely to cause death or significant injury.



**WARNING:** Indicates the existence of dangers that have the potential to cause death or significant injury.



**CAUTION:** Indicates the existence of dangers that have the potential to cause light or moderate injury.

**NOTICE:** Indicates the existence of dangers that may cause damage other than bodily harm.

Note that the “NOTE” header is used to indicate precautions about device handling and operation.

- Do not perform any operations not included in this manual. If a problem occurs with the device, contact maintenance personnel.
- Before using this device, read the safety precautions in this manual carefully to understand them.
- Keep this manual close at hand for easy reference.
- The precautions indicated on the device and in this manual have been given much consideration, but unforeseen cases may still occur. During operation, exercise personal caution in addition to following instructions.



## SAFETY SUMMARY (Continued)

---

### 1. GENERAL PRECAUTIONS

Read the following safety precautions carefully to understand them.

#### 1.1 WARNING

- This device was not designed for use with systems with requirements directly connected to human safety. Contact our sales staff if such usage may occur.
- In the unlikely event that use of this device causes smoke or strange odors, turn it off, unplug it, and contact sales staff or maintenance personnel. Continued use could result in fire or electric shock.
- Since this device contains a magnetic disk, take care not to jolt, shock, or shake it. In the unlikely event that this device is dropped or its case is damaged, unplug it and then contact maintenance personnel. Continued use could result in fire or electric shock. Likewise, avoid any shock during unpackaging and transport, as these can cause malfunctions.
- Do not physically alter this device, as it could result in fire or electric shock. Keep in mind that we take no responsibility for damage caused as a result of such alteration.
- To mitigate unlikely device malfunctions or faults due to failure or misuse, provide external protection and safety circuits, and design your system by taking care to implement protective measures against personal injury or disaster.



## SAFETY SUMMARY (Continued)

---

### 1.2 CAUTION

- Always leave space in front of and in back of this device for air circulation openings and for maintenance operation. Failure to do so can lead to high temperatures, causing damage or decreasing lifespan.
- This device may cause personal injury if it falls or tips over. Make sure that this device is appropriately fastened when carrying it.
- Keep take care to prevent your fingers from being caught or snapped in this device when unpacking or carrying it.



## SAFETY SUMMARY (Continued)

---

### 1.3 NOTICE

- This device cannot guarantee system safety by itself. To mitigate unlikely device malfunctions or faults due to failure or misuse, design your system by taking care to implement protective measures against personal injury or disaster, such as external protection and safety circuits.
- Use the basic software that we specify. Operation is not guaranteed if any other basic software is used.
- Avoid using this device in dusty areas or environments with corrosive gases, as this may lead to failure.
- When emergency cutoff is performed (the power cable is unplugged or the breaker is cut without performing OS shutdown processing), the system program may require reloading due to loss of journal information, depending on the processing program. As such, do not perform emergency cutoff unless absolutely necessary due to malfunction.
- Keep in mind that if the power supply is cut, the system may not be able to recover automatically.

## PREFACE

This manual explains how to use RAS (Reliability, Availability, Serviceability) feature for the Hitachi Industrial Computers HF-W2000 Model 35/30 (herein as “this device”).

### <Manual organization>

This manual consists of the following:

- CHAPTER 1 WHAT RAS FEATURE CAN DO
- CHAPTER 2 RAS FEATURE MONITORING TARGETS
- CHAPTER 3 SETTING UP RAS FEATURE
- CHAPTER 4 CHECKING HARDWARE STATUS
- CHAPTER 5 CONTROLLING HARDWARE
- CHAPTER 6 LIBRARY FUNCTIONS
- CHAPTER 7 MAINTENANCE AND FAILURE ANALYSIS
- APPENDIX A RUNNING THE COMMAND PROMPT AS AN ADMINISTRATOR

### <Precautions when using RAS feature>

- Extent of support for RAS feature  
Note that the supported RAS feature differs between the HF-W2000 Model 35 (herein as “Model 35”) and the HF-W2000 Model 30 (herein as “Model 30”). For details about the extent of support, see “CHAPTER 1 WHAT RAS FEATURE CAN DO.”
- RAS external control interfaces  
The explanations in this manual cover RAS external control interfaces, but these interfaces are optional, and can only be used with Model 35.
- Fast user-switching feature of Windows®  
The RAS feature explained in this manual does not support the fast user-switching feature of Windows®. The GUIs, libraries, and event objects described in this manual can only be used by the user who logged on first. To change the user who is using them, first log off all users, and then log on again.
- Event logs during SNMP service startup  
When remote notification feature is used, and the standard Windows® SNMP service is enabled, event ID 1500 may be recorded in the error log when the SNMP service starts up. This event log is recorded when SNMP trap notification settings have not been performed. Set up trap notifications as shown in “4.5.3 Procedures for starting remote notification feature.”



<Note for storage capacity calculations>

- Memory capacities and requirements, file sizes and storage requirements, etc. must be calculated according to the formula  $2^n$ . The following examples show the results of such calculations by  $2^n$  (to the right of the equals signs).

1 KB (kilobyte) = 1,024 bytes

1 MB (megabyte) = 1,048,576 bytes

1 GB (gigabyte) = 1,073,741,824 bytes

- As for disk capacities, they must be calculated using the formula  $10^n$ . Listed below are the results of calculating the above example capacities using  $10^n$  in place of  $2^n$ .

1 KB (kilobyte) = 1,000 bytes


1 MB (megabyte) =  $1,000^2$  bytes

1 GB (gigabyte) =  $1,000^3$  bytes

<Trademarks>

- Microsoft®, Windows®, Windows NT®, Visual C++®, and Visual Basic® are trademarks or registered trademarks of Microsoft Corporation in the United States and other countries.
- Other company or product (software or hardware) names used in this manual other than those above are trademarks, registered trademarks, or products of their respective properties.

# CONTENTS

 SAFETY SUMMARY .....	S-1
CHAPTER 1 WHAT RAS FEATURE CAN DO .....	1-1
CHAPTER 2 RAS FEATURE MONITORING TARGETS .....	2-1
2.1 Fan Monitoring Feature .....	2-1
2.2 Internal Cubicle Temperature Monitoring Feature .....	2-2
2.3 Drive Failure Prediction Feature (SMART Monitoring) .....	2-3
2.4 Drive Usage Time Monitoring Feature [Model 35 Only] .....	2-4
2.5 Memory Status Monitoring Feature .....	2-5
2.6 OS Lock Monitoring Feature [Model 35 Only] .....	2-6
2.7 Watchdog Timer Monitoring Feature [Model 35 Only] .....	2-8
2.7.1 Automatic retrigger feature for the watchdog timer .....	2-8
2.7.2 Monitoring the user program operation status .....	2-8
2.8 RAID Monitoring Feature [RAID Model Only] .....	2-10
2.8.1 RAID status transition .....	2-11
2.8.2 Media error .....	2-13
CHAPTER 3 SETTING UP RAS FEATURE .....	3-1
3.1 RAS Feature Settings Window .....	3-1
3.1.1 Overview .....	3-1
3.1.2 How to open the RAS feature settings window .....	3-2
3.1.3 How to use the RAS feature settings window .....	3-3
3.1.4 Editing popup notification messages .....	3-14
CHAPTER 4 CHECKING HARDWARE STATUS .....	4-1
4.1 Hardware Status Display Window .....	4-2
4.1.1 Overview .....	4-2
4.1.2 Hardware status display icon .....	4-3
4.1.3 Hardware status display window .....	4-6
4.2 RAS Event Notification Feature [Model 35 Only] .....	4-12
4.2.1 Overview .....	4-12
4.2.2 Event acquisition methods .....	4-12
4.2.3 Examples of event object usage .....	4-13
4.3 Popup Notification Feature [Model 35 Only] .....	4-14
4.3.1 Overview .....	4-14
4.3.2 Contents of displayed messages .....	4-15
4.3.3 Popup notification feature settings .....	4-15
4.4 Digital LED Feature for Status Indication [Model 35 Only] .....	4-16
4.4.1 Overview .....	4-16
4.4.2 Displayed status codes .....	4-17
4.4.3 Types of status display modes .....	4-19
4.4.4 List of hardware status codes .....	4-20
4.4.5 Digital LED for status indication control function .....	4-21
4.5 Remote Notification Feature [Model 35 Only] .....	4-22
4.5.1 Overview .....	4-22
4.5.2 Hardware statuses for which remote notification is performed .....	4-23
4.5.3 Procedures for starting remote notification feature .....	4-24

4.5.4	List of extension MIB objects for HF-W .....	4-30
4.5.5	HF-W extension MIB file .....	4-36
4.6	Status Acquisition Using the RAS Library [Model 35 Only] .....	4-37
<b>CHAPTER 5 CONTROLLING HARDWARE .....</b>		<b>5-1</b>
5.1	Automatic Device Shutdown .....	5-2
5.1.1	Automatic shutdown due to fan failure detection .....	5-2
5.1.2	Automatic shutdown due to high-temperature error detection .....	5-3
5.1.3	Automatic shutdown due to remote shutdown signal detection [Model 35 only] .....	5-3
5.2	Control by RAS Library [Model 35 Only] .....	5-4
5.3	RAID Configuration Control Command (raidctrl) [RAID Model Only] .....	5-5
<b>CHAPTER 6 LIBRARY FUNCTIONS .....</b>		<b>6-1</b>
6.1	RAS Library Interface [Model 35 Only] .....	6-1
6.1.1	Overview .....	6-1
6.1.2	Shutdown function (BSSysShut) .....	6-4
6.1.3	Watchdog timer control function (WdtControl) .....	6-5
6.1.4	Output control functions for general external controls (GendoControl, GendoControlEx) .....	6-9
6.1.5	Input status acquisition functions for general external controls (GetGendi, GetGendiEx) .....	6-12
6.1.6	Log information recording function (MConWriteMessage) .....	6-15
6.1.7	Memory status acquisition function (GetMemStatus) .....	6-17
6.1.8	Function for obtaining drive failure prediction information (GetHddPredict) .....	6-19
6.1.9	Drive status acquisition function (hfwDiskStat) .....	6-21
6.1.10	Functions for obtaining RAID status .....	6-23
6.1.11	Functions for controlling the digital LED for status indication .....	6-28
6.1.12	Sample programs .....	6-31
<b>CHAPTER 7 MAINTENANCE AND FAILURE ANALYSIS .....</b>		<b>7-1</b>
7.1	STOP Error Code Cause Notification Feature .....	7-1
7.1.1	Overview .....	7-1
7.1.2	Target STOP error causes .....	7-1
7.1.3	Event logs .....	7-2
7.2	Log Information Collection Window [Model 35 Only] .....	7-3
7.2.1	Overview .....	7-3
7.2.2	Opening the log information collection window .....	7-3
7.2.3	Using the log information collection window .....	7-4
7.2.4	Closing the log information collection window .....	7-5
7.3	Trend Log of Internal Cubicle Temperatures [Model 35 Only] .....	7-6
7.3.1	Overview .....	7-6
7.3.2	Log files .....	7-6
7.3.3	Command for setting the temperature log cycle .....	7-8
<b>APPENDIX A RUNNING THE COMMAND PROMPT AS AN ADMINISTRATOR .....</b>		<b>A-1</b>

## FIGURES

Figure 1-1	RAS Feature Settings Window .....	1-2
Figure 1-2	Hardware Status Display Icon .....	1-3
Figure 2-1	Status Lamp and CPUSTOP Contact Operation .....	2-7
Figure 2-2	Example of Monitoring for User Program Operation Status .....	2-8
Figure 2-3	RAID Status Transition for Model 35 .....	2-11
Figure 2-4	RAID Status Transition for Model 30 .....	2-12
Figure 2-5	Example of Message Box When a Media Error Occurred .....	2-13
Figure 3-1	RAS Feature Settings Window .....	3-1
Figure 3-2	Shutdown Setting .....	3-3
Figure 3-3	Watchdog Timer Setting .....	3-5
Figure 3-4	Drive Failure Prediction Setting (SMART) .....	3-7
Figure 3-5	Drive Usage Monitoring Setting .....	3-8
Figure 3-6	Detailed Settings for Drive Usage Time Monitoring Feature .....	3-9
Figure 3-7	Digital LED Setting .....	3-10
Figure 3-8	Popup Notification Setting .....	3-11
Figure 3-9	Detailed Settings for Popup Notification Feature .....	3-12
Figure 3-10	Definition File Format .....	3-14
Figure 4-1	Hardware Status Display Window .....	4-2
Figure 4-2	Example of Popup Description of an Icon (When the Hardware Status Is Normal) .....	4-5
Figure 4-3	Example of Popup Description of an Icon (When the Hardware Status Is an Error) .....	4-5
Figure 4-4	Hardware Status Display Icon Menu .....	4-5
Figure 4-5	How to Open the Hardware Status Display Window .....	4-6
Figure 4-6	Confirmation Message for Disconnection of the Drive in Drive Bay 1 .....	4-9
Figure 4-7	Reject Message for Disconnection of the Drive in Drive Bay 1 .....	4-10
Figure 4-8	Hardware Status Display Window (When Error Occurred) .....	4-11
Figure 4-9	Example of Popup Message Display .....	4-14
Figure 4-10	Digital LED for Status Indication .....	4-16
Figure 4-11	Hardware Status Codes .....	4-17
Figure 4-12	Application Status Codes .....	4-17
Figure 4-13	STOP Error Codes .....	4-18
Figure 4-14	Operation Example for Status Display Modes .....	4-19
Figure 6-1	WDTTO Contact Operation .....	6-8
Figure 6-2	WDTTO Contact Operation (During Shutdown) .....	6-8
Figure 6-3	GENDO Contact Operation .....	6-11
Figure 6-4	Log Information Format .....	6-15
Figure 7-1	Configuration of Collection Results Folders .....	7-5
Figure 7-2	Log Information Format 1 .....	7-7
Figure 7-3	Log Information Format 2 .....	7-7

## TABLES

Table 1-1	Overview of RAS Feature .....	1-1
Table 2-1	Device Status and MCALL Contact Operation for Fan Monitoring .....	2-1
Table 2-2	Device Status and MCALL Contact Operation for Internal Cubicle Temperature Monitoring .....	2-2
Table 2-3	Device Status and Status Lamp Operation for OS Lock Monitoring .....	2-6
Table 2-4	Device Status and CPUSTOP Contact Operation for OS Lock Monitoring .....	2-6
Table 2-5	RAID Status and MCALL Contact .....	2-10
Table 2-6	RAID Statuses for Model 35 and Their Meaning .....	2-11
Table 2-7	RAID Statuses for Model 30 and Their Meaning .....	2-12
Table 3-1	List of Items in the RAS Feature Settings Window .....	3-1
Table 3-2	Section Name and Defined Messages .....	3-15
Table 3-3	Contents of the [Object] List Box Corresponding to the Selected Event .....	3-16
Table 4-1	Hardware Status Display Icons .....	4-4
Table 4-2	Fan Status and Displayed Information .....	4-7
Table 4-3	Temperature Status and Displayed Information .....	4-7
Table 4-4	Drive Status and Displayed Information .....	4-8
Table 4-5	RAID Status and Information Displayed .....	4-10
Table 4-6	List of Reported Events .....	4-12
Table 4-7	List of Reported Events (For Legacy Program Compatibility) .....	4-13
Table 4-8	Displayed Message Contents .....	4-15
Table 4-9	Status Display Mode .....	4-19
Table 4-10	Hardware Status Code List .....	4-20
Table 4-11	List of Functions for Controlling the Digital LED for Status Indication .....	4-21
Table 4-12	Objects Related to Hardware Status .....	4-30
Table 4-13	Objects Related to RAS Feature Settings .....	4-32
Table 4-14	Objects Related to Operation Modes .....	4-32
Table 4-15	Objects Related to HF-W Extension MIBs .....	4-33
Table 4-16	Objects Related to Trap Notification (When Errors Occur) .....	4-34
Table 4-17	Objects Related to Trap Notification (During Recovery from Error) .....	4-35
Table 4-18	Objects Related to Trap Notification (Operation Mode) .....	4-35
Table 5-1	RAID Status Displayed by the raidctrl Command .....	5-6
Table 5-2	RAID Configuration Drive Status Displayed by the raidctrl Command .....	5-6
Table 5-3	Error Messages of the raidctrl Command .....	5-9
Table 6-1	List of Functions Provided by the RAS Library .....	6-1
Table 6-2	Processing Specified for dwCmd of WdtControl .....	6-5
Table 6-3	Processing Specified for dwCmd of the GendoControl Function .....	6-9
Table 6-4	Processing Specified by dwPort of the GendoControlEx Function .....	6-10
Table 6-5	Processing Specified by dwCmd of the GendoControlEx Function .....	6-10
Table 6-6	Processing Specified by dwPort of the GetGendiEx Function .....	6-13
Table 6-7	Values for Each Element of Dimm_Status .....	6-17
Table 6-8	Value for Each Element of the HDD_PREDICT_DATA Struct .....	6-19
Table 6-9	Values for Disk_Status .....	6-21
Table 6-10	Value for Level of the HFW_ARRAY_STATUS Struct .....	6-24
Table 6-11	Value for DiskNumber of the HFW_ARRAY_STATUS Struct .....	6-24
Table 6-12	Value for Status of the HFW_ARRAY_STATUS Struct .....	6-24
Table 6-13	Combination of Values for Status of the HFW_ARRAY_STATUS Struct .....	6-25
Table 6-14	Values for Elements of the RAID_DATA Struct .....	6-26
Table 6-15	Values Specified for dwMode of the SetMode7seg Function .....	6-30

Table 6-16 List of Provided Sample Programs ..... 6-31  
Table 7-1 List of Target STOP Error Causes ..... 7-1  
Table 7-2 Recorded Event Logs ..... 7-2  
Table 7-3 Recorded Log Files ..... 7-6

**This Page Intentionally Left Blank**

## CHAPTER 1 WHAT RAS FEATURE CAN DO

The HF-W series provides RAS (Reliability, Availability, Serviceability) feature through a high-reliability industrial computer.

The following gives an overview of RAS feature.

Table 1-1 Overview of RAS Feature

Yes: Supported, -: Unsupported

Category		Item	Model 35	Model 30
Monitoring feature		Device hardware monitoring	Yes (*1)	Yes (*1)
		OS lock monitoring	Yes	-
		Watchdog timer monitoring	Yes	-
GUI feature settings		RAS feature settings window	Yes	Yes (*2)
Status check	GUI display	Hardware status display window	Yes	Yes (*3)
	Notification feature	Event notification feature	Yes	-
		Popup notification feature	Yes	-
		Digital LED feature for status indication	Yes	-
		Remote notification feature	Yes	-
		Status acquisition by library functions	Yes	-
Control feature	Shutdown/Startup suppression	Automatic shutdown feature	Yes	Yes (*4)
		Startup suppression feature during major failure	Yes	Yes (*4)
		General external control	Yes	-
		Digital LED for status indication control	Yes	-
Library functions		RAS library	Yes	-
Maintenance /failure analysis	Memory dump-related	Memory dump collection feature	Yes	Yes
		STOP error code cause notification	Yes	-
		Log information collection window	Yes	-
		Maintenance operation support commands	Yes	-
		Internal cubicle temperature trend log	Yes	-

(\*1) Drive usage time monitoring feature is not supported for S model and Model 30.

(\*2) Feature not supported by Model 30 cannot be set up.

For details about unsupported feature, see Table 3-1 in “3.1 RAS Feature Settings Window.”

(\*3) Information about drive usage time is not displayed.

(\*4) Automatic shutdown and startup suppression is not performed by remote shutdown signal input detection.



## 1. WHAT RAS FEATURE CAN DO

### <Monitoring feature>

#### (1) Device hardware monitoring

This monitors the fan, internal cubicle temperature, drive status, etc. of this device.

#### (2) OS lock monitoring

This monitors the status of OS operation by using the OS operation monitoring timer implemented in this device. As long as top-level (realtime priority) processes can run normally, the status lamp on the front of this device is lit green.

#### (3) Watchdog timer monitoring

This monitors whether processes are scheduled normally by using the watchdog timer implemented in this device. It also provides a library for using watchdog timers.

### <GUI feature settings>

#### (4) RAS feature settings window

This allows conditions for performing automatically shutdown, watchdog timer usage, and other settings to be changed through graphical operation.

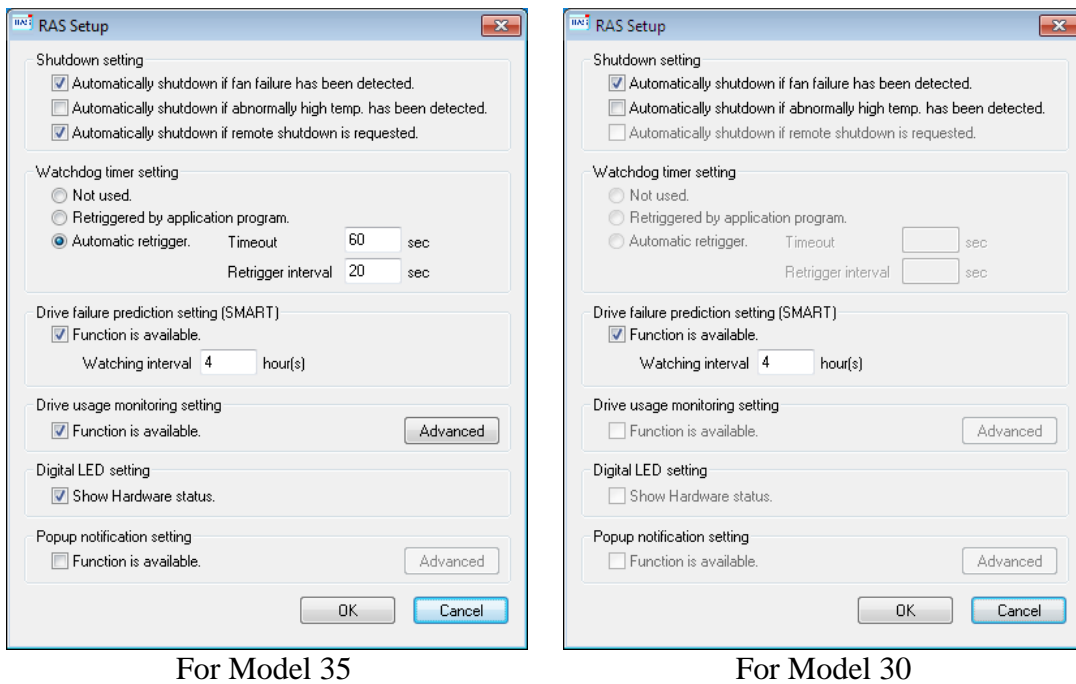


Figure 1-1 RAS Feature Settings Window

## &lt;Status confirmation&gt;

## (5) Hardware status display window

This displays the hardware status of this device in a graphical interface. It also has a resident icon for displaying the hardware status in the taskbar notification area.

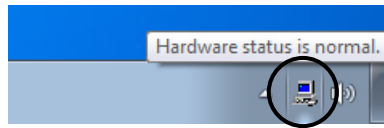


Figure 1-2 Hardware Status Display Icon

## (6) Event notification feature

This allows the hardware status to be selected by monitoring the status of event objects from user applications.

## (7) Popup notification feature

This performs notification by popup message when hardware errors occur in this device.

## (8) Digital LED feature for status indication

This reports hardware errors by means of the digital LED for status indication on the front of this device. Since this LED can also be used by user applications, it can also report application failure.

## (9) Remote notification feature

This allows the hardware status of this device to be checked from a remote environment. The remote environment can also be notified when the hardware status changes.

## (10) Status acquisition by library functions

The hardware status of this device can be obtained from user applications by using the RAS library.

## &lt;Control feature&gt;

## (11) Automatic shutdown feature

Automatic shutdown is performed when a fan failure, internal cubicle temperature error, or remote shutdown signal input is detected. Whether automatic shutdown is performed can be set on the screen of "(4) RAS feature settings window."

## (12) Startup suppression feature during major failure

This suppresses device startup to protect the hardware when a fan failure or other failure is detected during OS startup.

## (13) Control for general external controls and digital LED for status indication

The RAS library can be used to control the general external controls and digital LED for status indication.

General external controls include 4 contacts for input and 3 contacts for output that can be used by users. Through these contacts, signals are input from external devices to this device, and output from this device to external devices.

## 1. WHAT RAS FEATURE CAN DO

### <Library functions>

#### (14) RAS library interface

In addition to the library functions in (10) and (13), a library function for recording log information is provided.

### <Maintenance and failure analysis>

#### (15) Memory dump collection feature

When a failure such as unexpected stop occurs in this device, pressing the reset switch records the contents of system memory into a file (memory dump file). These memory dump contents can be analyzed to investigate the cause of failure.

#### (16) STOP error code cause notification

When a blue screen with the STOP error code 0x80 is detected, the cause of the blue screen is recorded in the event log.

#### (17) Log information collection window

This allows log information data and memory dump files for this device to be collected through graphical operations.

#### (18) Maintenance operation support commands

These commands include the one for saving memory dump files, event log files, and other failure information to external media.

#### (19) Internal cubicle temperature trend log

This performs regular acquisition of the internal cubicle temperature of this device, and records it in a file.

This manual explains the feature for (1) through (11), (13), (14), (16), (17), and (19). For details about the other features, refer to the *HF-W2000 Model 35/30 INSTRUCTION MANUAL* (manual number WIN-62-0060).

## CHAPTER 2 RAS FEATURE MONITORING TARGETS

This chapter explains the monitoring targets of RAS feature.

For details about the hardware specifications for (and purpose of each contact of) the RAS external control interface (option) explained in this chapter, refer to the *HF-W2000 Model 35/30 INSTRUCTION MANUAL* (manual number WIN-62-0060).

### 2.1 Fan Monitoring Feature

Fan monitoring feature monitors the statuses of multiple fans mounted in this device, and performs notification using the following when an error occurs in any fan:

- (1) Hardware status display window
- (2) Event notification feature
- (3) Popup notification feature
- (4) Digital LED feature for status indication
- (5) Remote notification feature
- (6) Automatic shutdown feature
- (7) Lit alarm lamp

For details about (1) to (5), see “CHAPTER 4 CHECKING HARDWARE STATUS.” For details about (6), see “5.1 Automatic Device Shutdown.”

Also, if the MCALL contact of the RAS external control interface is used, errors can be detected for this device through external hardware connected to the RAS external interface. The operation of the MCALL contact is as follows.

Table 2-1 Device Status and MCALL Contact Operation for Fan Monitoring

Status of this device	MCALL contact
During device startup or power off	Open
When the fan is normal or recovering from error	Open
Fan failure	Close

Since the alarm lamp (and the MCALL contact of the RAS external control interface) is used other than for fan monitoring, it does not turn off (open) if an error is detected on another monitoring target even if the fan failure is recovered.

**2.2 Internal Cubicle Temperature Monitoring Feature**

Internal cubicle temperature monitoring feature uses the temperature sensor within this device to monitor the status of the internal cubicle temperature, and performs notification using the following when an error occurs due to high internal cubicle temperature:

- (1) Hardware status display window
- (2) Event notification feature
- (3) Popup notification feature
- (4) Digital LED feature for status indication
- (5) Remote notification feature
- (6) Automatic shutdown feature
- (7) Lit alarm lamp

For details about (1) to (5), see “CHAPTER 4 CHECKING HARDWARE STATUS.” For details about (6), see “5.1 Automatic Device Shutdown.”

Also, if the MCALL contact of the RAS external control interface is used, errors can be detected for this device through external hardware connected to the RAS external interface. The operation of the corresponding contact is as follows.

Table 2-2 Device Status and MCALL Contact Operation for Internal Cubicle Temperature Monitoring

Status of this device	MCALL contact
During device startup or power off	Open
When normal or recovering from temperature error	Open
Temperature error	Close

Since the alarm lamp (and the MCALL contact of the RAS external control interface) is used other than for internal cubicle temperature monitoring, it does not turn off (open) if an error is detected on another monitoring target even if the temperature error is recovered.

### 2.3 Drive Failure Prediction Feature (SMART Monitoring)

The drive of this device comes with a failure prediction feature (SMART: Self-Monitoring, Analysis and Reporting Technology), which enables constant monitoring of the drive operation status to predict failure before it occurs. Drive failure prediction feature performs notification using the following when a drive may fail in the immediate future:

- (1) Hardware status display window
- (2) Event notification feature
- (3) Popup notification feature
- (4) Digital LED feature for status indication
- (5) Remote notification feature
- (6) The hfwDiskStat function and GetHddPredict function of RAS library

For details about (1) to (6), see “CHAPTER 4 CHECKING HARDWARE STATUS.”

This feature can be enabled/disabled in the RAS feature settings window. If it is disabled, notification is not performed by the above features. For details, see “3.1.3 How to use the RAS feature settings window.”

---

#### NOTE

---

- Since SMART cannot predict all drive failures, drives may malfunction before SMART predict it.
  - When a drive failure is predicted, back up the data and replace the drive as soon as possible. For details about how to replace a drive, refer to the *HF-W2000 Model 35/30 INSTRUCTION MANUAL* (manual number WIN-62-0060).
  - This feature performs monitoring for internal drives recognized at OS startup. If a new drive is connected or a drive is replaced due to maintenance, it may take some time to recognize the drive during initial startup after it is connected, preventing it from being recognized as a monitoring target. In this case, restart the device.
-

### 2.4 Drive Usage Time Monitoring Feature **[Model 35 Only]**

Drive usage time monitoring feature accumulates the drive usage time of this device, and if the usage time exceeds the predetermined value, notification is performed as follows. This makes it easy to understand when to replace the drive for preventing failures related to drive lifetime.

- (1) Hardware status display window
- (2) Event notification feature
- (3) Popup notification feature
- (4) Remote notification feature
- (5) The hfwDiskStat function of RAS library

For details about (1) to (5), see “CHAPTER 4 CHECKING HARDWARE STATUS.”

This feature can be enabled/disabled in the RAS feature settings window. If it is disabled, notification is not performed by the above features. For details, see “3.1.3 How to use the RAS feature settings window.”

---

#### NOTE

---

- This feature is not supported for S model. Even if this function is enabled in the RAS feature setting window, monitoring cannot be made correctly. Leave the setting invalid.
  - This feature monitors the time of OS operation from when the OS starts up until it shuts down. As such, if the OS isn't running, the usage time is not monitored.
  - This feature has a drive management information file, which contains serial numbers (drive-specific information) for the mounted drives. If drives other than those whose serial numbers are recorded in the drive management information file, the cumulative usage time value for the corresponding drive is automatically reset.
  - This feature does not predict drive failure, but in the interests of preventative maintenance to extend product lifespan, we recommend replacing the drive when its usage time exceeds the predetermined value. For details about the drive replacement procedures, refer to the *HF-W2000 Model 35/30 INSTRUCTION MANUAL* (manual number WIN-62-0060).
  - This feature performs monitoring for internal drives recognized at OS startup. If a new drive is connected or a drive is replaced due to maintenance, it may take some time to recognize the drive during initial startup after it is connected, preventing it from being recognized as a monitoring target. In this case, restart the device.
-

## 2.5 Memory Status Monitoring Feature

Since this device contains memory with ECC (Error Checking and Correcting), any single-bit errors that occur in memory are automatically corrected without affecting device operation.

However, if memory error correction occurs frequently, since this may be due to memory failure, we recommend replacing memory modules as a preventative measure.

Memory status monitoring feature performs notification as follows when memory error correction occurs frequently:

- (1) Event notification feature
- (2) Popup notification feature
- (3) Remote notification feature
- (4) The GetMemStatus function of RAS library

For details about (1) to (4), see “CHAPTER 4 CHECKING HARDWARE STATUS.”

---

### NOTE

---

For details about replacing memory modules, refer to the *HF-W2000 Model 35/30 INSTRUCTION MANUAL* (manual number WIN-62-0060).

---



**2.6 OS Lock Monitoring Feature [Model 35 Only]**

OS lock monitoring feature uses an OS operation monitoring timer within this device to detect an occurrence of a situation where top-level (realtime priority) processes cannot be run because of kernel panic or a busy CPU due to drivers, and notifies such situation (called OS lock) with a status lamp on the front of the device.

Table 2-3 Device Status and Status Lamp Operation for OS Lock Monitoring

Status of this device	Status lamp
During device startup or power off	Red
When OS lock has not occurred or during recovery from OS lock	Green
When OS lock occurred	Red

Also, if the CPUSTOP contact of the RAS external control interface is used, the existence of OS lock can be detected through external hardware connected to the RAS external interface. The corresponding contact operation is as follows.

Table 2-4 Device Status and CPUSTOP Contact Operation for OS Lock Monitoring

Status of this device	CPUSTOP contact
During device startup or power off	Close
When OS lock has not occurred or during recovery from OS lock	Open
When OS lock occurred	Close

---

**NOTE**

- This feature treats a situation in which processes with realtime priority cannot run as an OS lock.
  - If an OS lock occurs, processes on the OS cannot run as scheduled, causing delays in processing and possibly affecting the equipment using this device. It is necessary to resolve the cause of the issue as quickly as possible.
  - When the OS starts up and this feature is started, the status lamp turns green.
  - When the OS shuts down and this feature is stopped, the status lamp turns red. However, since shutdown processing has not yet completed when the status lamp turns red, take care not to cut the power.
-

Figure 2-1 shows the status lamp and CPUSTOP contact operation.

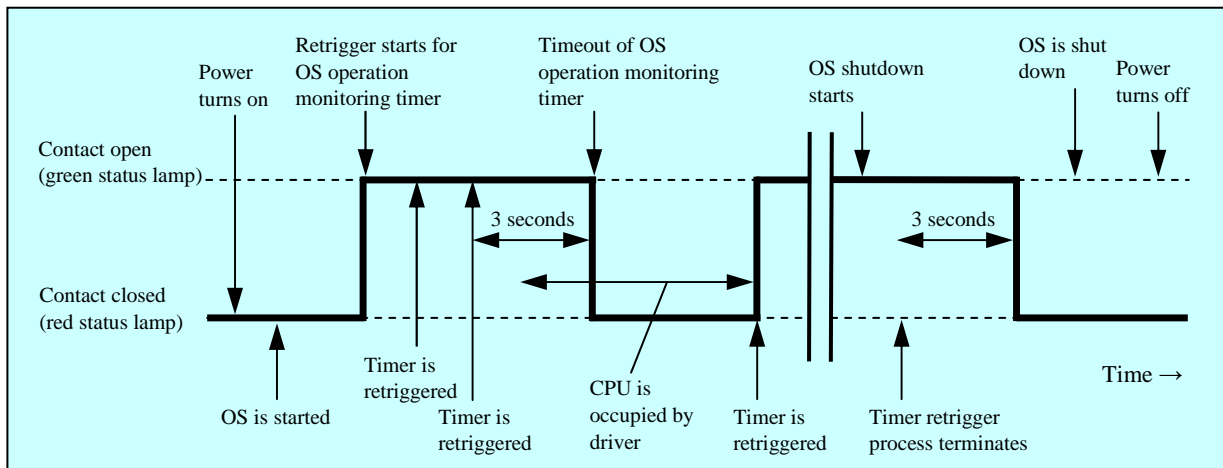


Figure 2-1 Status Lamp and CPUSTOP Contact Operation

The dotted lines in Figure 2-1 show each status of the corresponding contact, and the bold lines indicate the transition of the corresponding contact statuses.

## 2. RAS FEATURE MONITORING TARGETS

### 2.7 Watchdog Timer Monitoring Feature **[Model 35 Only]**

This device contains a watchdog timer which is automatically retriggered by this feature to monitor whether processes are being normally scheduled. Special library functions can also be used to monitor the operation status of user programs.

#### 2.7.1 Automatic retrigger feature for the watchdog timer

This feature consists of a process that regularly retriggeres the watchdog timer. Since this process runs at the lowest level (idle priority), a watchdog timer timeout occurs when any processes with a priority other than idle priority occupy the CPU longer than the set value. This allows runaway application processes to be detected.

The watchdog timer timeout time and the interval between retriggeres of this feature are set in the RAS feature settings window. For details about how to use the RAS feature settings window, see “3.1.3 How to use the RAS feature settings window.” The factory default settings are as follows:

Timeout time	60 (sec)
Retrigger interval	20 (sec)

#### 2.7.2 Monitoring the user program operation status

When the watchdog timer is used to monitor the operation status of user programs, the watchdog timer is regularly retriggered (the remaining time until watchdog timer timeout is reset to the initial value) by a user program, and the timeout check processing for the watchdog timer is performed from another program. The processing flow in this case is as follows.

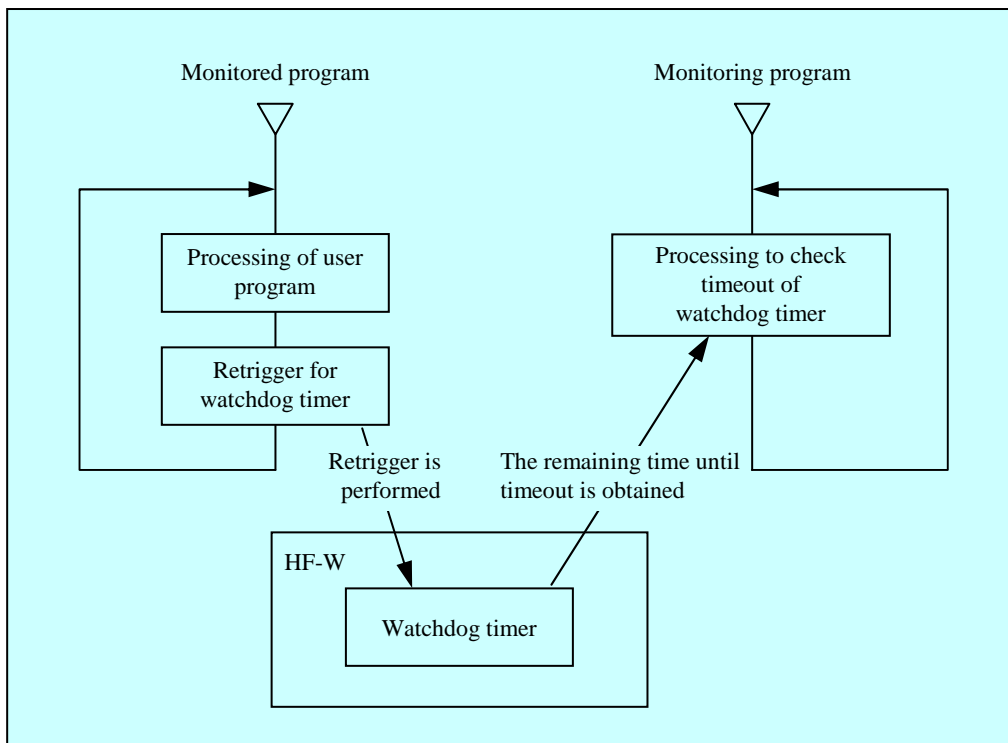


Figure 2-2 Example of Monitoring for User Program Operation Status

In Figure 2-2, the monitored target program regularly retriggers the watchdog timer. The monitoring program also regularly obtains the remaining time until watchdog timer timeout, and if the remaining time until timeout reaches 0, determines that a timeout has occurred.

The occurrence of a timeout indicates that the monitored program could not retrigger the watchdog timer for a duration longer than the set timeout time. In this example, timeout is detected for the watchdog timer by another program, but if the RAS external control interface is used, the WDTTO contact changes to the closed status when a timeout occurs for the watchdog timer. In this case, the operation status of user programs can be monitored from external hardware connected to the RAS external control interface.

When the watchdog timer is used from a program, the WdtControl library function is called. For details about how to use the WdtControl function, and WDTTO contact operation for the RAS external control interface, see “6.1.3 Watchdog timer control function (WdtControl).”

---

### NOTE

---

When the WdtControl function is used, automatic watchdog timer retrigger feature cannot be used. In the RAS feature settings window, select [Retriggered by application program] in the “Watchdog timer setting” area. For details about how to use the RAS feature settings window, “3.1.3 How to use the RAS feature settings window.”

---

**2.8 RAID Monitoring Feature [RAID Model Only]**

The RAID model is provided with the RAID1 function (hereinafter referred to as RAID) which is commonly known as mirror disk (mirroring).

The RAID monitoring feature monitors the RAID status of this device. When the RAID status changes, it notifies the user or application in the following methods.

- (1) Hardware status display window
- (2) Event notification feature [Model 35 only]
- (3) Popup notification feature [Model 35 only]
- (4) Digital LED feature for status indication [Model 35 only]
- (5) Remote notification feature [Model 35 only]
- (6) Lit alarm lamp
- (7) The hfwDiskStat, hfwRaidStat, and RaidStat functions of RAS library [Model 35 only]
- (8) The RAID configuration control command raidctrl

For details about the methods (1) to (5) above, see “CHAPTER 4 CHECKING HARDWARE STATUS.”

For details about (7), see “6.1.10 Functions for obtaining RAID status,” and for (8), see “5.3 RAID Configuration Control Command (raidctrl) [RAID Model Only].”

The MCALL contact of the RAS external contacts interface is also available to check against abnormality of this device via the external hardware connected to the RAS external contacts interface. The MCALL contact indicates the RAID status as below:

Table 2-5 RAID Status and MCALL Contact

RAID status	MCALL contact
Normal	Open
Degradation, Failure, Unknown	Close

The alarm lamp (and the MCALL contact of the RAS external contacts interface) is also used for other purposes than the RAID status monitoring. Therefore, if another error is detected by any other monitoring function at the same time, the lamp will not go off (or the contact will not open) even though the RAID status recovers to normal.

2.8.1 RAID status transition

(1) RAID status transition for model 35

Table 2-6 shows the RAID statuses for model 35 and their meaning, and Figure 2-3 shows its status transition.

Table 2-6 RAID Statuses for Model 35 and Their Meaning

RAID status		Description
	Detailed info	
Normal	–	Normal redundant operation.
Degradation	–	Redundancy is lost because of failure of one drive.
	Under rebuild	RAID is under rebuild due to replacement of a drive, etc.
Failure	–	Two drives have failed or no drive is connected.
Unknown	–	Raid status failed to be obtained.

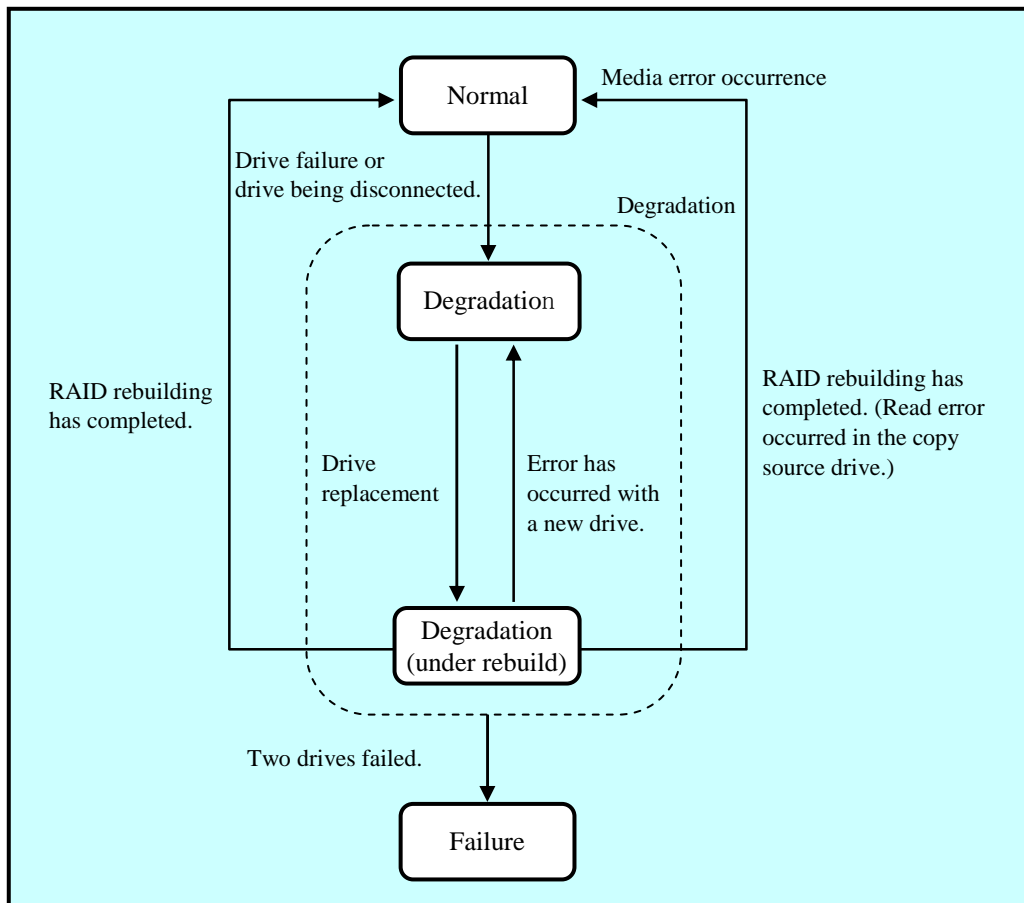


Figure 2-3 RAID Status Transition for Model 35

2. RAS FEATURE MONITORING TARGETS

(2) RAID status transition for model 30

Table 2-7 shows the RAID statuses for model 30 and their meaning, and Figure 2-4 shows its status transition.

Table 2-7 RAID Statuses for Model 30 and Their Meaning

RAID status		Description
	Detailed info	
Normal	–	Normal redundant operation.
Degradation	–	Redundancy is lost because of failure of one drive.
Unknown	–	RAID status failed to be obtained.

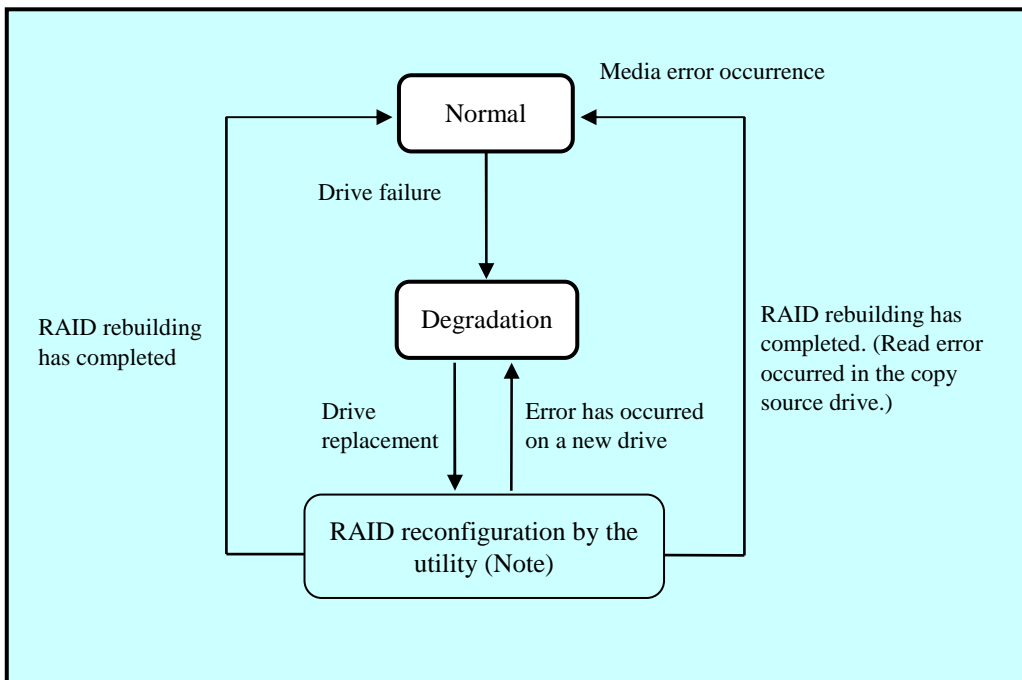


Figure 2-4 RAID Status Transition for Model 30

Note: In the case of model 30, use the RAID1 configuration utility that comes with the product to rebuild RAID1. For how to use the software RAID1 configuration utility, refer to the *HF-W2000 Model 35/30 INSTRUCTION MANUAL* (manual number WIN-62-0060).

### 2.8.2 Media error

A media error means the state where data completeness cannot be ensured even though the RAID status is normal. If a read error occurs while reading the copy source drive for rebuilding, the RAID rebuilding can finish normally but the data of the sector which failed in reading is lost, causing a lack of data completeness.

#### <Treatment of media error>

Whether lack of data completeness is treated as an error depends on whether the sector causing a read error is used. If a read error occurs in a sector in use, lack of data completeness may cause a problem. But if an error occurs in an unused sector, it causes no problem. This device is provided with a function that can determine whether a media error should be treated as an error according to the system operation. This function is available with the RAID configuration control command `raidctrl`. (For details, see “5.3 RAID Configuration Control Command (`raidctrl`) [RAID Model Only].”) Media errors are not treated as errors by default. If you determine to treat them as errors, they are notified as RAID status errors in the following methods.

- Event notification feature
- Digital LED feature for status indication
- Remote notification feature
- Lit alarm lamp
- MCALL contact of the RAS external contacts interface
- Message box display

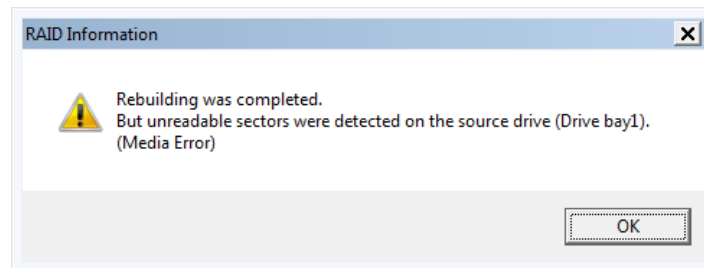


Figure 2-5 Example of Message Box When a Media Error Occurred

Regardless of media error treatment setting, you can always check occurrence of media errors with the event log, the `hfwRaidStat` function of the RAS library, and the RAID configuration control command `raidctrl`.



**This Page Intentionally Left Blank**

## CHAPTER 3 SETTING UP RAS FEATURE

### 3.1 RAS Feature Settings Window

#### 3.1.1 Overview

The RAS feature settings window can be used to set the following features:

Table 3-1 List of Items in the RAS Feature Settings Window

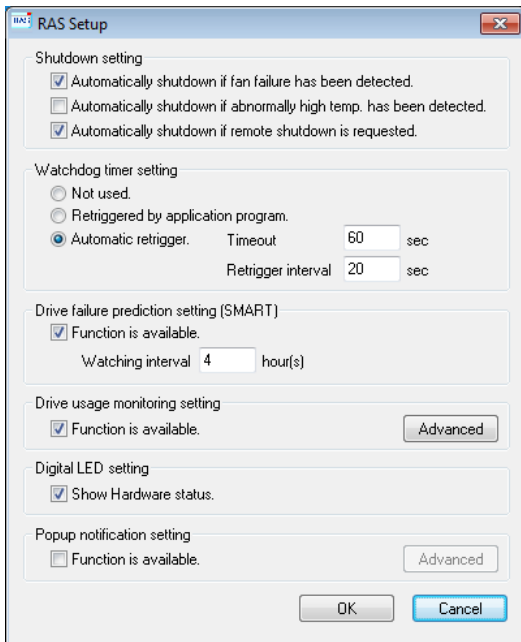
Yes: Supported, -: Unsupported

Item		Model 35	Model 30
Shutdown setting	Automatically shutdown if fan failure has been detected.	Yes	Yes
	Automatically shutdown if abnormally high temp. has been detected.	Yes	Yes
	Automatically shutdown if remote shutdown is requested.	Yes	-
Watchdog timer setting		Yes	-
Drive failure prediction setting (SMART)		Yes	Yes
Drive usage monitoring setting		Yes	-
Digital LED setting		Yes	-
Popup notification setting		Yes	-

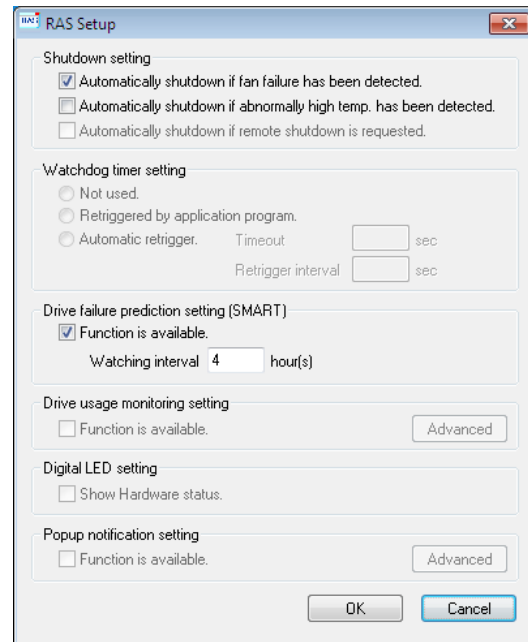
**NOTE**

Items not supported by Model 30 are always disabled, and cannot be enabled.

Figure 3-1 shows the RAS feature settings window. The following figure shows the factory default settings for the device.



For Model 35



For Model 30

Figure 3-1 RAS Feature Settings Window

### 3. SETTING UP RAS FEATURE

#### 3.1.2 How to open the RAS feature settings window

The following explains how to open the RAS feature settings window.

To use this window, first log on as an administrator account for the computer.

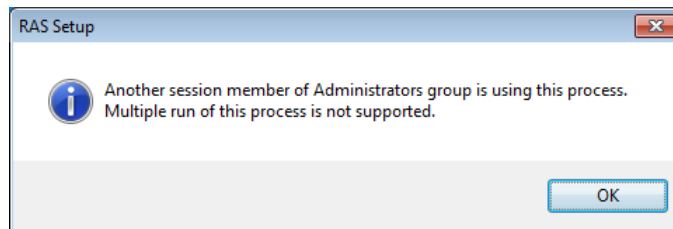
1. Click the [Start] button.
2. Choose [All programs], and then [RAS Software].
3. Click [RAS Setup].
4. If the [User Account Control] window is displayed, click the [Yes] button.  
(For Windows® 7 only)

---

#### NOTE

---

The RAS feature settings window cannot be opened by multiple users at the same time. As such, when multiple consoles attempt to open this window by means of user switching, etc., the following dialog box is displayed. In this case, close of this window running on another console, and then open the RAS feature settings window again.



## 3.1.3 How to use the RAS feature settings window

## (1) Shutdown feature settings

This sets whether shutdown is performed automatically in the case of fan failures, high-temperature errors, or remote shutdown input.

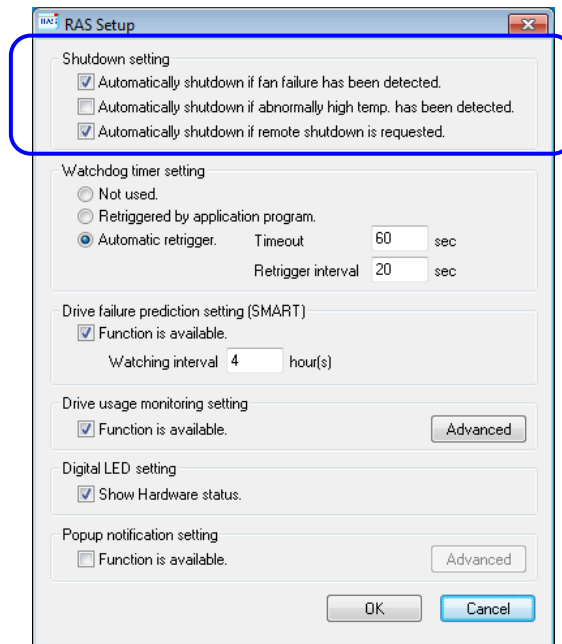


Figure 3-2 Shutdown Setting

- [Automatically shutdown if fan failure has been detected.] checkbox
  - Selected: Perform automatic shutdown (factory default setting)
  - Cleared: Do not shutdown
  
- [Automatically shutdown if abnormally high temp. has been detected.] checkbox
  - Selected: Perform automatic shutdown
  - Cleared: Do not shutdown (factory default setting)
  
- [Automatically shutdown if remote shutdown is requested.] checkbox
  - Selected: Perform automatic shutdown (factory default setting)
  - Cleared: Do not shutdown

To change the current settings, click the corresponding checkbox.

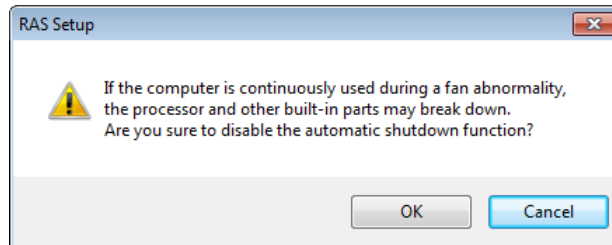
### 3. SETTING UP RAS FEATURE

---

#### NOTE

---

- If use of this device is continued while a fan failure exists, the processor and other built-in parts may not be cooled sufficiently, causing runaway system processing and parts damage due to device malfunction. As such, when the [Automatically shutdown if fan failure has been detected.] checkbox is cleared, the following warning dialog box is displayed. Leave this feature enabled (selected) whenever possible.



If the [Cancel] button is clicked in the above dialog box, the checkbox returns to the selected status. If the [OK] button is clicked, the checkbox is cleared.

- When this feature is used to perform automatic shutdown, the power is cut after shutdown processing.
  - Since remote shutdown input detection is performed every 5 seconds, it may take up to 5 seconds for shutdown processing to actually start after remote shutdown input is performed.
-

## (2) Watchdog timer settings

This sets the watchdog timer included in this device.

For watchdog timer usage, any of the following can be selected by clicking the appropriate radio button:

- Not used.
- Retriggered by application program.
- Automatic retrigger.

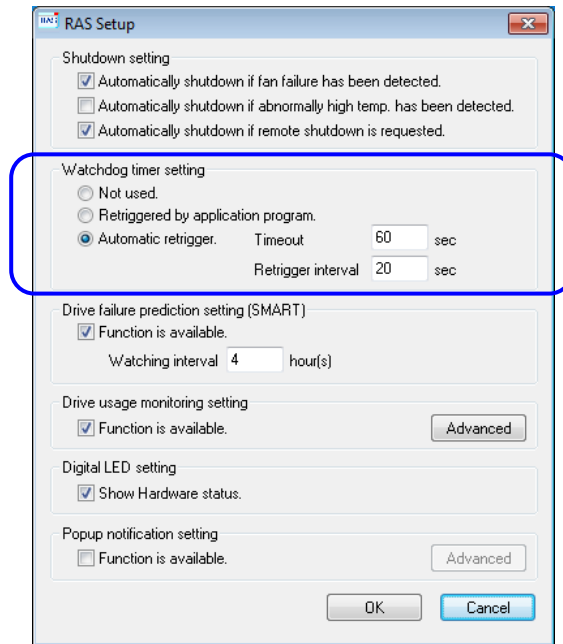


Figure 3-3 Watchdog Timer Setting

[Not used.]:

When this item is selected, the watchdog timer is stopped, and does not time out. The watchdog timer cannot be used from the WdtControl function of the RAS library.

[Retriggered by application program.]:

When this item is selected, the user application can use the WdtControl function of the RAS library to control the watchdog timer.

---

NOTE

When the setting is changed to [Retriggered by application program.], the watchdog timer is stopped temporarily, during which it will not time out. The countdown for the watchdog timer is started when a user application uses the WdtControl function to retrigger the watchdog timer. The status of the watchdog timer (counting down or stopped) can be obtained using the WdtControl function.

---

### 3. SETTING UP RAS FEATURE

[Automatic retrigger.]:

When this item is selected, watchdog timer automatic retrigger feature is used to automatically retrigger the watchdog timer. In this case, the watchdog timer cannot be used from the WdtControl function of the RAS library.

When this item is selected, the retrigger interval of watchdog timer automatic retrigger feature and the watchdog timer timeout time can be set.

- Timeout

This sets the timeout time for the watchdog timer. This can be set as an integer from 5 to 60 (in seconds). The factory default setting is 60 (seconds).

- Retrigger interval

This sets the retrigger interval for automatic retrigger feature. This can be set as an integer from 1 to the timeout time minus 4 (in seconds). The factory default setting is 20 (seconds). Note that the retrigger interval can only be entered after the timeout time entry.

The timeout time and retrigger interval can be set only when [Automatic retrigger.] is selected, and cannot be entered otherwise. The factory default setting for the watchdog timer setting is [Automatic retrigger.].

## (3) Drive failure prediction feature (SMART monitoring) setting

This sets the failure prediction feature (SMART monitoring) for the internal drive of this device.

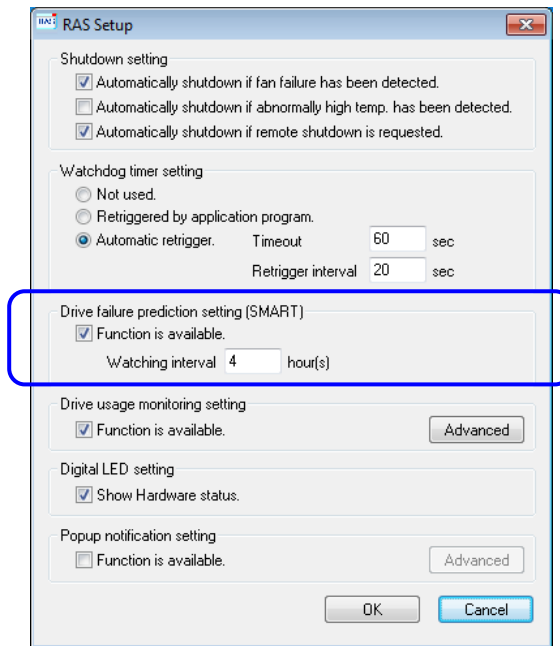


Figure 3-4 Drive Failure Prediction Setting (SMART)

- [Function is available.] checkbox
  - Selected: Drive failure prediction feature is enabled (factory default setting)
  - Cleared: Drive failure prediction feature is disabled

To change the current setting, click the checkbox.

If drive failure prediction feature is disabled, features (such as drive failure prediction status display) described in “2.3 Drive Failure Prediction Feature (SMART Monitoring)” cannot be used.

- Watching interval
 

The watching interval time can be set in the box to the right of “Watching interval”. Integers from 1 to 24 can be set (in hours). The factory default setting is 4 (hours). If drive failure prediction feature is disabled, this item cannot be set.



### 3. SETTING UP RAS FEATURE

#### (4) Drive usage time monitoring feature setting

This performs settings for drive usage time monitoring feature. The [Advanced] button allows detailed settings to be defined for this feature.

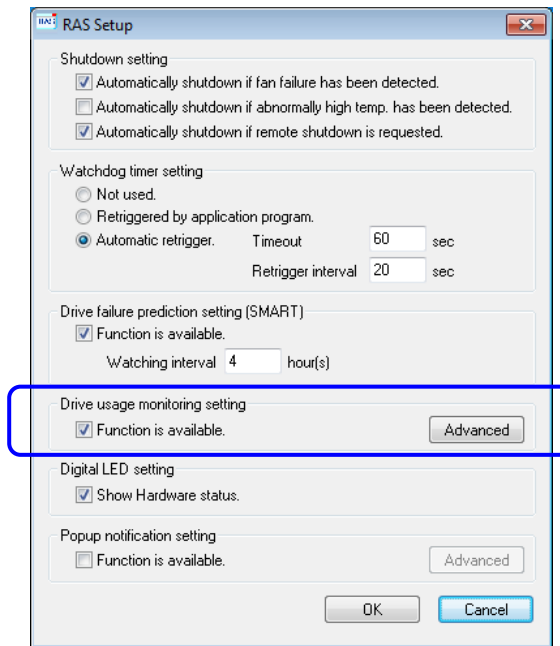


Figure 3-5 Drive Usage Monitoring Setting

- [Function is available.] checkbox
  - Selected: Drive usage time monitoring feature is enabled (factory default setting)
  - Cleared: Drive usage time monitoring feature is disabled

To change the current setting, click the checkbox.

If drive usage time monitoring feature is disabled, features (such as drive usage overtime display) described feature shown in “2.4 Drive Usage Time Monitoring Feature [Model 35 Only]” cannot be used.

---

#### NOTE

Drive usage time monitoring feature is not supported for S model of Model 35. Even if this function is enabled, monitoring cannot be made correctly. Leave the checkbox unselected.

---

- [Advanced] button

Click the [Advanced] button to display the window shown below.

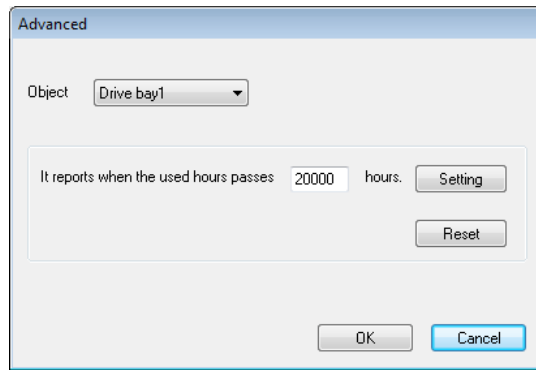
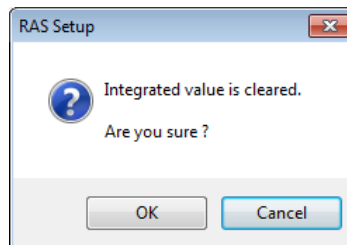


Figure 3-6 Detailed Settings for Drive Usage Time Monitoring Feature

The [Advanced] window can be used to set the time at which exceeded drive usage time is reported. To set the notification time, select the drive for which the value is to be set from the appropriate list box, enter a numeric value in the [It reports when the used hour passes  hours.], and click the [Setting] button.

The number of hours can be set from a minimum of 100 to a maximum of 20000 in increments of 100 (100 (minimum value), 200, ..., 20000 (maximum value)). The factory default setting is 20000 (hours). To clear the current integrated value after drive replacement, etc., click the [Reset]. In this case, the following confirmation dialog box is displayed.



If the [OK] button is clicked in the above dialog box, the current integrated value is cleared. If the [Cancel] button is clicked, the integrated value is not cleared. Note that if drive usage time monitoring feature is disabled, this item cannot be set.

If it is okay to perform detailed setting changes and save the changed settings, click the [OK] button.

Otherwise, to leave the settings as they are, click the [Cancel] button. This window will close, and the changed settings will not be reflected.

### 3. SETTING UP RAS FEATURE

#### (5) Digital LED for status indication setting

This sets the display mode of the digital LED for status indication mounted on the front of the device.

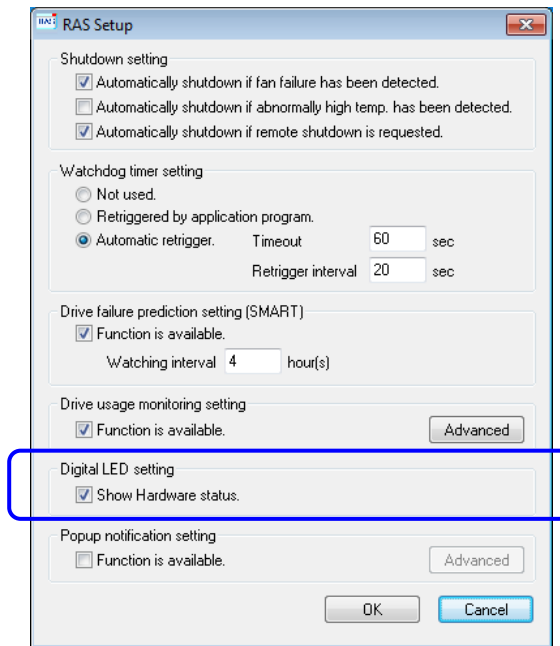


Figure 3-7 Digital LED Setting

- [Show hardware status.] checkbox
  - Selected: Hardware status display mode (factory default setting)
  - Cleared: Application status display mode

To change the current setting, click the checkbox.

For details about digital LED feature for status indication such as the display mode for the digital LED for status indication, see “4.4 Digital LED Feature for Status Indication [Model 35 Only].”

## (6) Popup notification feature setting

This sets up popup notification feature. The [Advanced] button allows detailed settings to be defined for this feature.

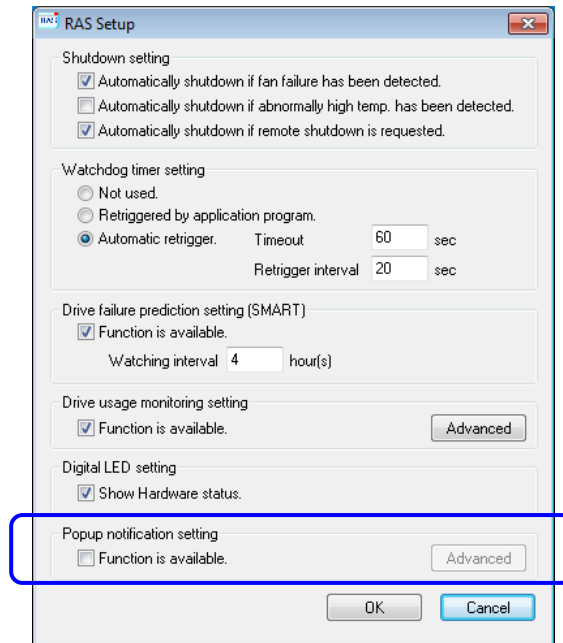


Figure 3-8 Popup Notification Setting

- [Function is available.] checkbox
  - Selected: Popup notification feature is enabled
  - Cleared: Popup notification feature is disabled (factory default setting)

To change the current setting, click the checkbox.

If popup notification feature is disabled, popup notification described in “4.3 Popup Notification Feature [Model 35 Only]” cannot be performed.

### 3. SETTING UP RAS FEATURE

- [Advanced] button

Click the [Advanced] button to display the window shown below.

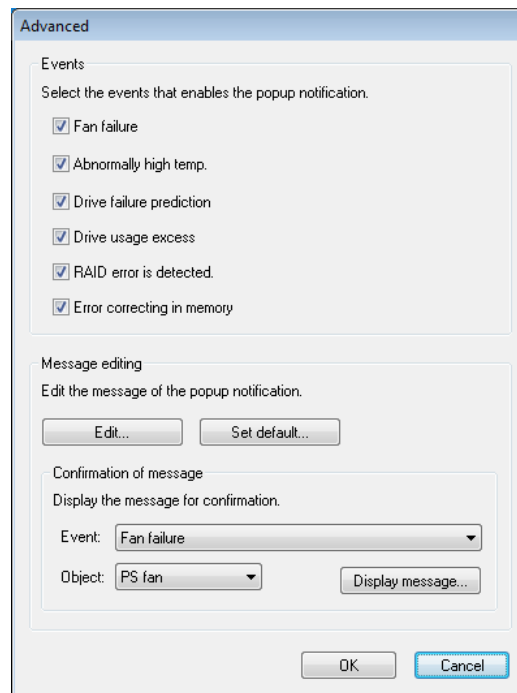


Figure 3-9 Detailed Settings for Popup Notification Feature

#### [Events]

- Fan failure
- Abnormally high temp.
- Drive failure prediction
- Drive usage excess
- RAID error is detected. (RAID model only)
- Error correcting in memory

Set whether or not to perform popup notification for each of the above events.

- Checkboxes for each event

- Selected: Popup notification is performed (factory default setting).
- Cleared: Popup notification is not performed.

To change the current setting, click the checkbox.

#### [Message editing]

The contents of popup notification messages can be edited. The edited results can also be confirmed. For details about how to edit and confirm messages, see “3.1.4 Editing popup notification messages.”

If it is okay to perform detailed setting changes and save the changed settings, click the [OK] button.

Otherwise, to leave the settings as they are, click the [Cancel] button. This window will close, and the changed settings will not be reflected.

(7) Enabling/disabling setting changes in (1) to (6)

If it is okay to enable the settings changed in (1) to (6), click the [OK] button. The RAS feature settings window will close, and the setting changes will take effect.

To leave the settings for (1) to (6) as they are, click the [Cancel] button. The RAS feature settings window will close, and the changed settings will not be reflected.

### 3. SETTING UP RAS FEATURE

#### 3.1.4 Editing popup notification messages

##### (1) How to edit popup notification messages

To edit the messages displayed in popup notifications, click the [Edit...] button. Notepad is opened with the message definition file for the popup notification. Edit the message according to the format, and then save it.

---

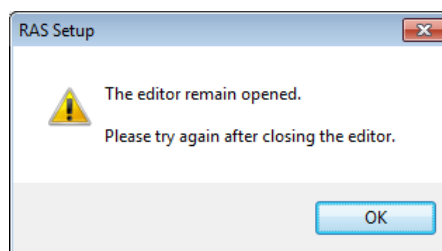
#### NOTE

---

The following operations cannot be performed while a message is being edited:

- Clicking the [Edit...] button
- Clicking the [Set default...] button
- Closing the RAS feature settings window (by clicking the [OK] button or [Cancel] button)

If any of the above operations are performed, the following warning dialog box is displayed.



Click the [OK] button in the above dialog box to return to the RAS feature settings window.

---

#### ■ Definition file format

The following shows the format for definition files.

```
;- sample of message --  
[PS-FAN] ← Section  
Line1=""  
Line2="Revolutions of fan at power supply deteriorated"  
Line3="remarkably"  
← Key ← Value  
[CPU-FAN]  
Line1=""  
Line2=""  
Line3="Revolutions of cpu fan deteriorated remarkably."
```

Figure 3-10 Definition File Format

A definition file consists of sections, keys, and their values.

A section contains keys and values, with each key and value separated by an equals sign (=).

Semicolons (;) indicate comment lines.

## ■ Definition file description

### 1. Section

The following explains the section names that can be defined for this feature and messages to be defined.

Table 3-2 Section Name and Defined Messages

Section name	Defined message
[PS-FAN]	Message displayed when power fan failure is detected
[CPU-FAN]	Message displayed when CPU fan failure is detected
[TEMP]	Message displayed when internal cubicle temperature error is detected
[HDD1-SMART]	Message displayed when failure prediction (SMART) is detected for the drive in drive bay 1
[HDD2-SMART]	Message displayed when failure prediction (SMART) is detected for the drive in drive bay 2
[HDD1-OFFLINE]	Message displayed when an RAID error due to drive error in drive bay 1 is detected. (*)
[HDD2-OFFLINE]	Message displayed when an RAID error due to drive error in drive bay 2 is detected. (*)
[HDD1-OVERRUN]	Message displayed when the usage time exceeds the predetermined value for the drive in drive bay 1
[HDD2-OVERRUN]	Message displayed when the usage time exceeds the predetermined value for the drive in drive bay 2
[DIMM1-ERR]	Message displayed when DIMMA error correction is detected frequently
[DIMM2-ERR]	Message displayed when DIMMB error correction is detected frequently

(\*) RAID model only

### 2. Keys

This specifies the line number of the message displayed in the popup message. For this feature, line 1 through line 5 can be set in each section. Any keys set on line 6 or after are ignored.

### 3. Values

This specifies a line of message text to be displayed in the popup message. Up to 50 bytes of characters can be specified for each key. If more than 50 bytes of characters are defined, the 51st and subsequent bytes are ignored. To specify whitespace characters, wrap the entire value in double quotation marks (""). A key with no value is treated as a line change.

---

## NOTE

- When saving edited contents, make sure to use “Overwrite Save”. Other operations may not reflect the edited contents properly.
  - During editing operation, do not use any other applications to edit definition files. When a definition file is edited by multiple applications, the edited contents may not be reflected properly.
  - Make sure that popup notification message clearly indicates that an error has occurred. Continued operation despite the fact that an error has occurred may significantly affect the system.
-



### 3. SETTING UP RAS FEATURE

#### (2) Confirming the results of message editing

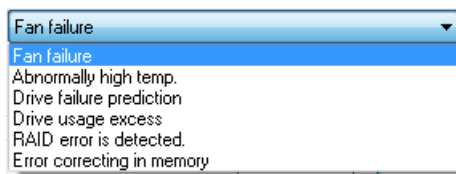
The results of message editing can be confirmed for each of the following events.

- Fan failure
- Abnormally high temp.
- Drive failure prediction
- Drive usage excess
- RAID error is detected. (RAID model only)
- Error correcting in memory

The following shows how to confirm the results of message editing.

1. From the [Event] list box, select an event to be confirmed.

This list box displays only events for which the [Events] checkbox is selected. Note that if none of the [Events] checkboxes are selected, this list box cannot be used.



<Example for fan failure selection>

2. From the [Object] list box, choose an object to be confirmed.

The contents of this list box differ depending on the event selected in step 1.



<Example for PS fan selection>

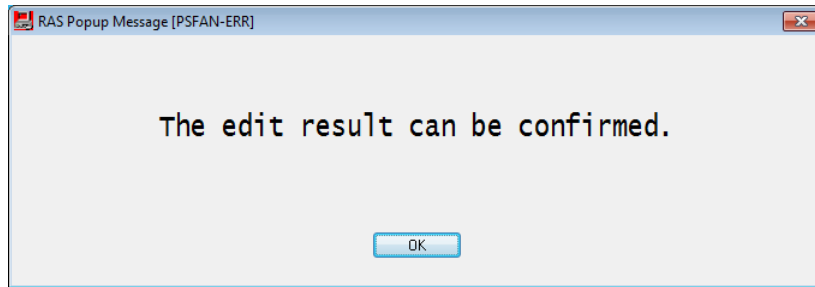
The following shows the contents of the [Object] list box corresponding to the event selected from the [Event] list box.

Table 3-3 Contents of the [Object] List Box Corresponding to the Selected Event

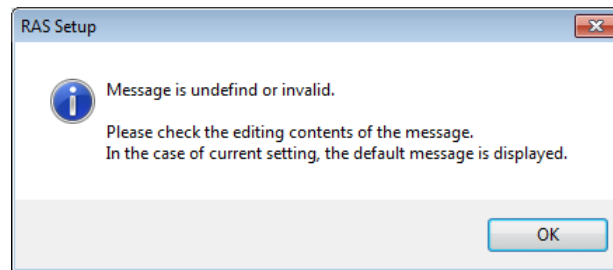
[Event] list box selection	[Object] list box contents
Fan failure	PS fan or CPU fan
Abnormally high temp.	Internal temp.
Drive failure prediction	Drive bay1 or Drive bay2
Drive usage excess	
RAID error is detected.	
Error correcting in memory	DIMM A or DIMM B

## 3. Click the [Display message...] button.

A popup message appears reflecting the edited message contents. After confirmation, click the [OK] button in the popup window.

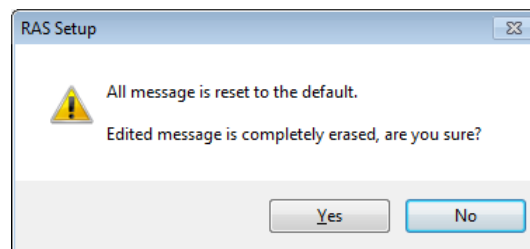


If the message has not been edited, or if the message definition file contents were specified incorrectly, the following dialog box is displayed. Click the [OK] button to return to the RAS feature settings window, and correct the edited message contents.



## (3) Revert to the default message

To revert to the default popup notification message, click the [Set default...] button. The following dialog box is displayed, so click the [Yes] button. The edited message definition file contents are deleted.



Click the [No] button to leave the edited contents intact without reverting the message to the default value.

**This Page Intentionally Left Blank**

## CHAPTER 4 CHECKING HARDWARE STATUS

The hardware status of this device can be checked as follows.

(1) Confirmation by GUI

The hardware status of this device can be checked using a graphical interface. For details, see “4.1 Hardware Status Display Window.”

(2) Confirmation by user application

The event object status can be monitored from a user application, allowing the hardware status of this device to be checked. For details, see “4.2 RAS Event Notification Feature [Model 35 Only].”

The hardware status of this device can also be obtained from a user application by using the RAS library. For details, see “4.6 Status Acquisition Using the RAS Library [Model 35 Only].”

(3) Confirmation by desktop screen of this device

Popup messages can be used to perform notification when a hardware error occurs in this device. For details, see “4.3 Popup Notification Feature [Model 35 Only].”

(4) Confirmation by digital LED for status indication on the front of this device

The digital LED for status indication on the front of this device can be used to indicate an occurrence of a hardware error of this device. This LED can also be used by a user application to report failure to maintenance personnel. For details, see “4.4 Digital LED Feature for Status Indication [Model 35 Only].”

(5) Confirmation from remote environments

The hardware status of this device can be checked from a remote environment, and changes in hardware status can be reported to remote environments. For details, see “4.5 Remote Notification Feature [Model 35 Only].”

## 4. CHECKING HARDWARE STATUS

### 4.1 Hardware Status Display Window

#### 4.1.1 Overview

When logon is performed on this device, an icon indicating the hardware status appears and remains in the taskbar notification area. Detailed information about the hardware status of this device can be displayed by double-clicking this icon, or right-clicking it and choosing [Display Hardware status] from the displayed popup menu.

The information displayed in this window is as follows.

- Fan status
- Internal cubicle temperature status
- Drive failure prediction (SMART monitoring) status
- Drive usage time
- RAID status (RAID model only)

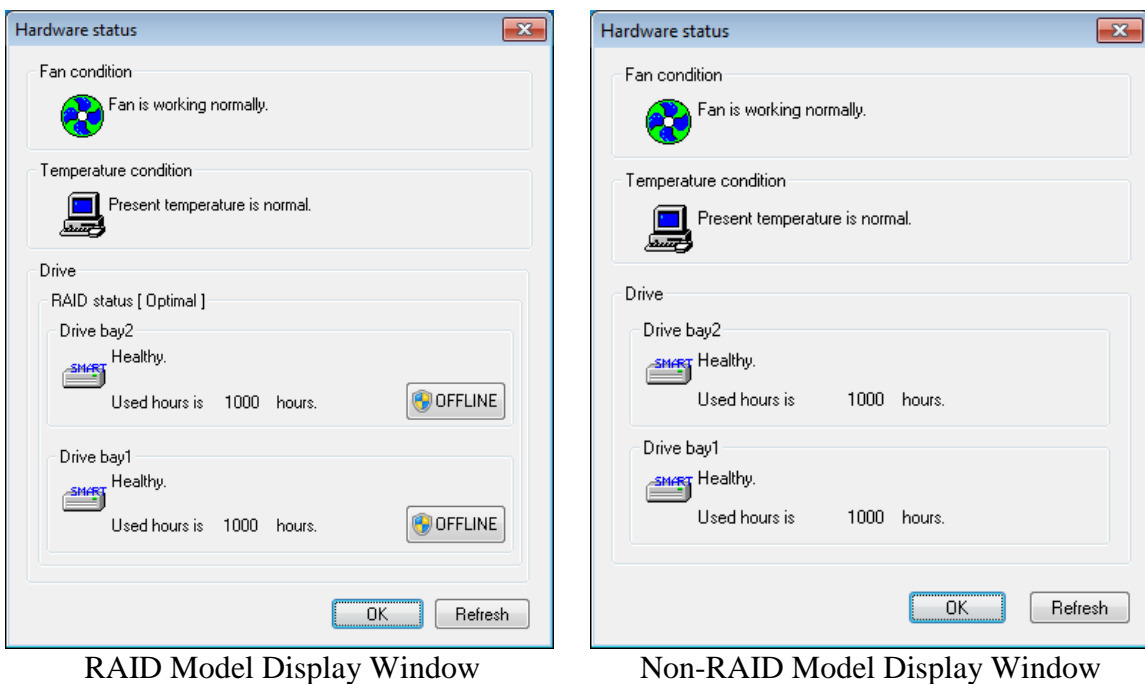


Figure 4-1 Hardware Status Display Window

---

#### NOTE

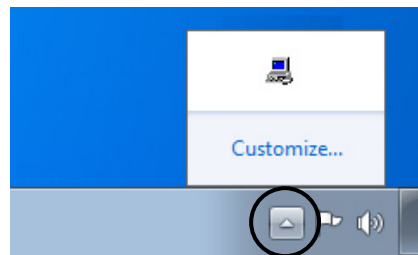
- S model and model 30 do not support drive usage time monitoring feature. As such, no icon and information for drive usage time is displayed. Model 30 does not support offline operation. The [OFFLINE] button is therefore disabled.
  - The drives for which this window is displayed are internal drives recognized at OS startup. If a new drive is connected or replaced due to maintenance, it may take some time to recognize the drive during initial startup after the connection, preventing drive-related information from being displayed. In this case, restart this device.
  - Act quickly to resolve any problems with hardware for which errors are displayed in this window. For details about replacement procedures to improve product lifespan, refer to the *HF-W2000 Model 35/30 INSTRUCTION MANUAL* (manual number WIN-62-0060).
-

## 4.1.2 Hardware status display icon

When logon is performed for this device, an icon indicating hardware status appears and remains in the taskbar notification area.



For Windows® 7, no icon is displayed in the taskbar notification area by default, but the arrow on the side of the notification area can be clicked to display the icon. Then, [Customize...] can be clicked to display the icon in the taskbar notification area.

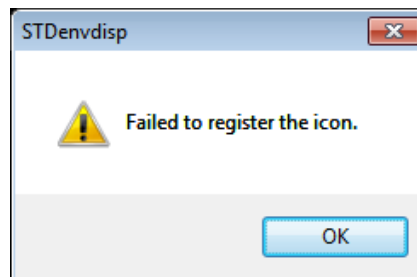



---

 NOTE
 

---

In rare cases, the hardware status display icon may fail to be registered, and the following dialog box is displayed. In this case, perform the following procedures to re-execute the hardware status display icon.





1. In the above dialog box, click the [OK] button.
  2. From [Start], choose [All Programs] and then [Startup], and click [RAS Status].
-

#### 4. CHECKING HARDWARE STATUS

(1) Displayed icons and popup descriptions

Table 4-1 lists the displayed icons and their descriptions. The icon descriptions are displayed by mousing over the icons.

Table 4-1 Hardware Status Display Icons

No.	Hardware status	Icon	Icon description
1	Normal		Hardware status is normal.
2	Error		Fans failed.
3			Abnormal TEMP detected.
4			Fans failed. Abnormal TEMP detected.
5			Fans failed. Drive failure possible.
6			Abnormal TEMP detected. Drive failure possible.
7			Fans failed. Abnormal TEMP detected. Drive failure possible.
8			Fans failed. Time of drive exceeded.
9			Abnormal TEMP detected. Time of drive exceeded.
10			Fans failed. Abnormal TEMP detected. Time of drive exceeded.
11			RAID error detected.
12			Fans failed. RAID error detected.
13			Abnormal TEMP detected. RAID error detected.
14			Fans failed. Abnormal TEMP detected. RAID error detected.
15			Caution
16	Time of drive exceeded.		

No. 5 to 7, 15: If the drive usage time exceeds the predetermined value while drive failure is predicted, excess of drive usage time is not displayed as the icon description.

No. 8 to 10, 16: Displayed only for Model 35.

No. 11 to 14: The icon is displayed only for RAID models. If drive failure prediction or excess in drive usage time occurred at the same time as drive disconnection from RAID (offline status), drive failure prediction or excess in drive usage time is not shown as description for the icon.

No. 15, 16: If this is detected at the same time as a hardware status error, the hardware status error icon is displayed.

Figure 4-2 and Figure 4-3 give examples of popup descriptions of icons when the hardware status of this device is normal and when it is an error, respectively.

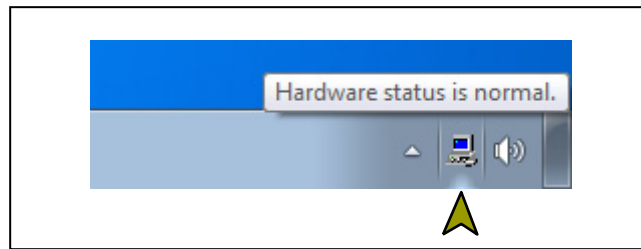


Figure 4-2 Example of Popup Description of an Icon (When the Hardware Status Is Normal)

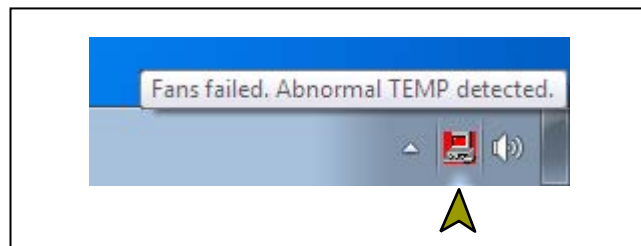


Figure 4-3 Example of Popup Description of an Icon (When the Hardware Status Is an Error)

- (2) Hardware status display icon menu  
 Right-click the icon to display a popup menu.

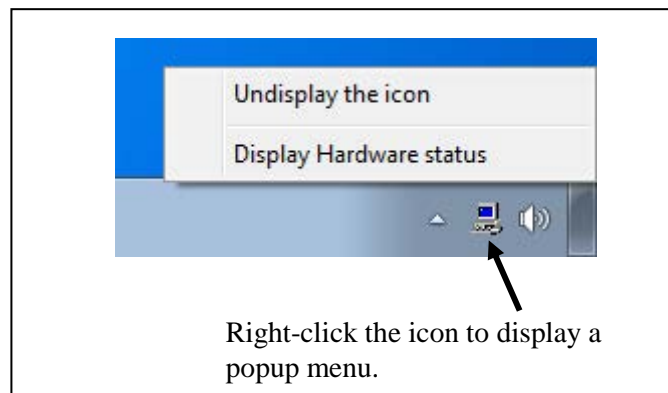


Figure 4-4 Hardware Status Display Icon Menu

- [Display Hardware status]  
 Click this to display the hardware status display window.
- [Undisplay the icon]  
 Click this to delete the icon from the taskbar notification area.



## 4. CHECKING HARDWARE STATUS

### 4.1.3 Hardware status display window

The hardware status display window displays information about the hardware status of this device.

Figure 4-5 shows how to open the hardware status display window.

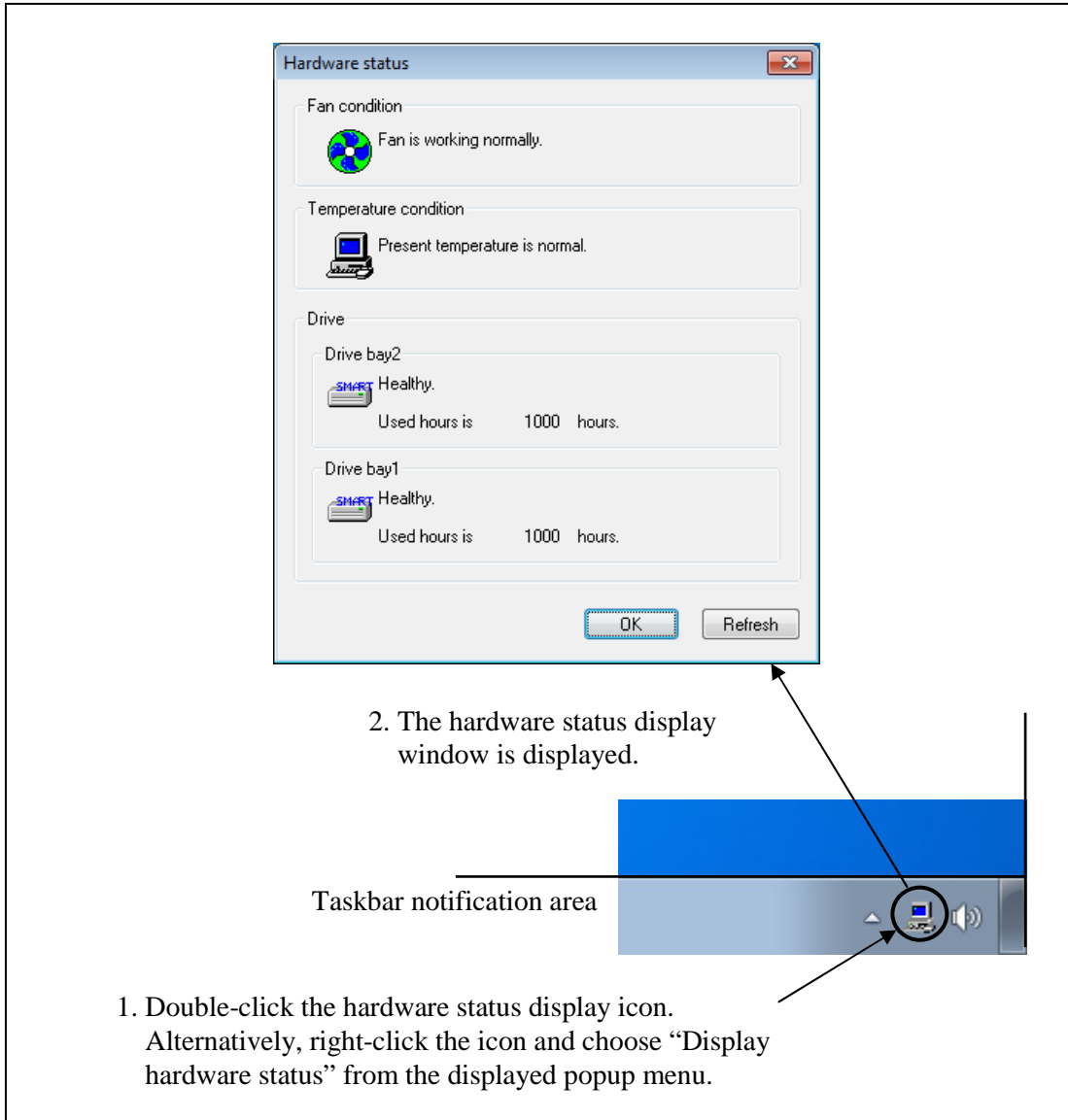




Figure 4-5 How to Open the Hardware Status Display Window

## (1) Description of the window

## 1. Fan status

The current fan status is displayed.



Table 4-2 Fan Status and Displayed Information

Fan status	Icon	Information
Normal		Fan is working normally.
Low revolution count		Fan failure is detected. For details, refer to the event log.

## 2. Temperature status

The current status of the internal cubicle temperature is displayed.

Table 4-3 Temperature Status and Displayed Information










Temperature status	Icon	Information
Normal		Present temperature is normal.
High-temperature error		Temperature in the unit has exceeded the upper limit.

#### 4. CHECKING HARDWARE STATUS

##### 3. Drive status

The current status of the drive is displayed. The drive statuses of drive bay 1 and drive bay 2 are displayed in the lower area and in the upper area, respectively.

Table 4-4 Drive Status and Displayed Information

No.	Drive status		Icon	Information
1	Normal	SMART monitoring is enabled		Healthy.
2		SMART monitoring is disabled		SMART is not available.
3	Failure prediction by SMART			A failure may be imminent.
4	Usage time exceeding predetermined value	SMART monitoring is enabled		Used hours exceeded prescribed value.
5		SMART monitoring is disabled		Used hours exceeded prescribed value.
6	Error	Offline		OFFLINE.
7				OFFLINE.(Incomplete data) Please rebuild.
8		Drive under rebuild		REBUILD.
9		Unknown		Unknown.
10	Unmounted			Not Connected.

No. 6 to 9: Displayed only for RAID model.

No. 4, 5, 8: Displayed only for Model 35.

No.7: Displayed only for model 30. A data mismatch occurred between drives and the drive is disconnected from RAID because power failure occurred or the Windows® did not shutdown normally. The drive can be used continuously by performing rebuild unlike in the case of the offline status in No. 6.

##### 4. Drive usage time display [Model 35 only]

The usage time (current integrated value) of the corresponding drive is displayed. The integrated value is updated every hour from 0 to 100 hours, and every 10 hours from 100 hours and up. The range that can be displayed for integrated values is 0 to 99990 (hours). If drive usage time monitoring feature is disabled, the usage time is not displayed.

## 5. [Refresh] button

This obtains the latest hardware status information, and updates the displayed information.

## 6. [OK] button

This closes the hardware status display window.

---

**NOTE**

---

- When drive failure is predicted to occur, drive hardware failure may occur in the near future. We recommend you perform data backup and replace the drive.
  - Drive usage time monitoring feature is not supported for Model 30. As such, the following information is not displayed.
    3. Excess of usage time of the drive status (No. 4 and No. 5 in Table 4-4)
    4. Drive usage time display
- 

## 7. [OFFLINE] button [RAID model only]

This disconnects the drive from RAID.

This button is available when both drives of RAID are normal and when failure prediction by SMART and excess in drive usage time are shown.

The administrator privilege is required to disconnect the drive. Log on to the computer using the administrator account and click the [OFFLINE] button.

After you click the [OFFLINE] button, a confirmation message appears as shown in Figure 4-6. To disconnect the drive from RAID and bring it offline, click the [Yes] button. Clicking [No] cancels the [OFFLINE] button action.

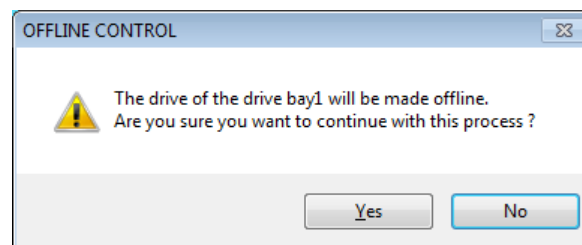


Figure 4-6 Confirmation Message for Disconnection of the Drive in Drive Bay 1

#### 4. CHECKING HARDWARE STATUS

##### NOTE

- This button is unavailable with Model 30.
- The disconnected drive cannot be used as an online drive again. Before disconnecting a drive, make sure that you are going to disconnect the correct one.
- In Windows® 7, if the user who does not have administrator privilege clicks the [OFFLINE] button while the user account control (UAC) is disabled, the following message box appears and the drive will not be disconnected. To disconnect the drive, log on using the administrator account before click the [OFFLINE] button.

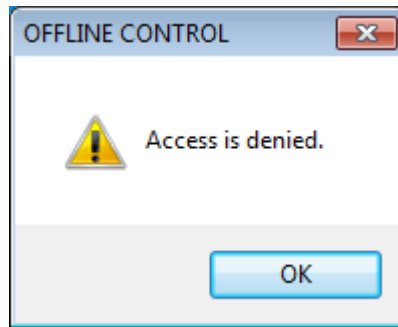


Figure 4-7 Reject Message for Disconnection of the Drive in Drive Bay 1

- This button is unavailable in a remote session.

#### 8. RAID status [RAID model only]

This indicates the current RAID status.

Table 4-5 RAID Status and Information Displayed

No.	RAID status		Information displayed	Remarks	
		Detailed info			
1	Normal	-	[ Optimal ]		
2			[ Optimal (Media Error) ]	Media error occurred.	
3	Degradation	-	[ Degrade ]		
4			[ Degrade (Media Error) ]		
5			Under rebuild	[ Degrade (Rebuild: xx%) ]	xx: Progress in rebuilding
6				[ Degrade (Rebuild: xx%, Media Error) ]	xx: Progress in rebuilding Media error occurred.
7	Failure	-	[ Fail ]	This may not be displayed because the system stops.	
8	Unknown	-	[ Unknown ]		

No. 5, 6, and 7: Displayed only for Model 35.

Figure 4-8 shows the hardware status display window when a hardware status error occurred. (The window shows the fan status error, abnormal temperature, failure prediction (SMART) detected in the drive of drive bay 1, and offline status of the drive in drive bay 2.)

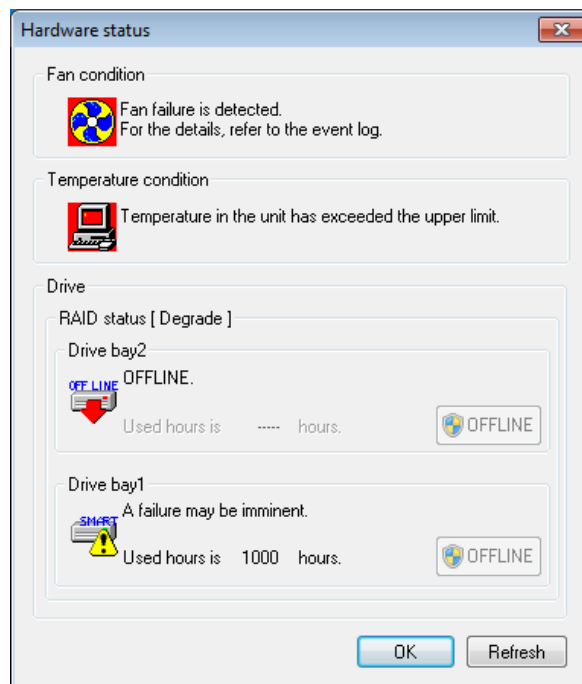


Figure 4-8 Hardware Status Display Window (When Error Occurred)

## 4. CHECKING HARDWARE STATUS

### 4.2 RAS Event Notification Feature [Model 35 Only]

#### 4.2.1 Overview

This feature can notify applications of occurrence of hardware or other errors of which the user should be notified, by setting the event object to a signal status.

Applications can monitor for when the event object becomes a signal status, to detect when hardware or other errors occur.

Note that event object signal statuses are reset when their cause is resolved.

---

#### NOTE

---

This feature is not supported for Model 30.

---

#### 4.2.2 Event acquisition methods

An event occurrence can be detected as follows.

1. The OpenEvent function of the Windows API can be used to obtain an event object handle.

In this case, specify SYNCHRONIZE for the parameter indicating access type (dwDesiredAccess).

2. The WaitForSingleObject function or WaitForMultipleObject function of the Windows API can be used to monitor for whether the corresponding event object has become a signal status.

Table 4-6 lists the events to be reported to users and their event objects.

Table 4-6 List of Reported Events

No.	Event	Event object name
1	PS fan failure	W2KRAS_PSFAN_ERR_EVENT
2	CPU fan failure	W2KRAS_CPUFAN_ERR_EVENT
3	Internal cubicle temperature error	W2KRAS_TEMP_ERR_EVENT
4	Remote shutdown request	W2KRAS_RMTSTDN_EVENT
5	Failure predicted on either drive	W2KRAS_HDD_PREDICT_EVENT
6	Predetermined usage time value exceeded on either drive	HFW_HDD_OVERRUN_EVENT
7	Error correction occurred in either memory with high frequency	HFW_MEMORY_ERR_EVENT
8	RAID status is normal	HFW_RAID_OPTIMAL_EVENT
9	RAID status is not normal	HFW_RAID_DEGRADE_EVENT

No. 5: This includes cases in which drive failure prediction information could not be obtained.

No. 8, 9: Supported only for RAID model

No. 9: After the RAID rebuilding has completed, the signal status changes to the non-signal status.

In addition to the objects described above, the following event objects are provided in Windows® 7 (32-bit) in order to maintain backward compatibility with the Windows NT® programs and conventional equipment.

Table 4-7 List of Reported Events (For Legacy Program Compatibility)

No.	Event	Event object name
1	PS fan failure	SMGRASBD_PSFAN_ERR_EVENT
2	CPU fan failure	SMGRASBD_CPUFAN_ERR_EVENT
3	Internal cubicle temperature error	SMGRASBD_TEMP_ERR_EVENT
4	Remote shutdown request	SMGRASBD_RMTSTDN_EVENT
5	Failure predicted on a drive	SMGSMART_HDD_PREDICT_EVENT
6	Predetermined usage time value exceeded on the drive in drive bay 1	HFW_HDD1_OVERRUN_EVENT
7	Predetermined usage time value exceeded on the drive in drive bay 2	HFW_HDD2_OVERRUN_EVENT
8	RAID status is normal	HFW_RAID_ONLINE_EVENT
9	RAID status is not normal	HFW_RAID_OFFLINE_EVENT

No. 5: This includes cases in which drive failure prediction information could not be obtained.

No. 8, 9: Supported only for RAID model

No. 9: After the RAID rebuilding has completed, the signal status changes to the non-signal status.

---

NOTE

- For Windows® 7, “Global\” needs to be prepended to the event object name.
  - The event objects in Table 4-7 are not provided in Windows® 7 (64-bit). Use the event objects in Table 4-6 instead.
- 

#### 4.2.3 Examples of event object usage

A sample C program for monitoring event objects is provided. For detail about the location and file name for the sample program, see “6.1.12 Sample programs.”



## 4. CHECKING HARDWARE STATUS

### 4.3 Popup Notification Feature [Model 35 Only]

#### 4.3.1 Overview

This feature can notify users of hardware errors or other events that should be reported to users occur, by displaying popup messages on the desktop screen. This enables users to know when hardware errors and other events occur.

Specifically, popup messages are displayed in the following cases.

- Fan failure
- Internal cubicle temperature errors
- Drive failure is predicted
- Drive usage time exceeds the predetermined value
- Memory error correction occurs with high frequency
- RAID error is detected (RAID model only)

Figure 4-9 gives an example of popup message notification for when power fan failure occurs.

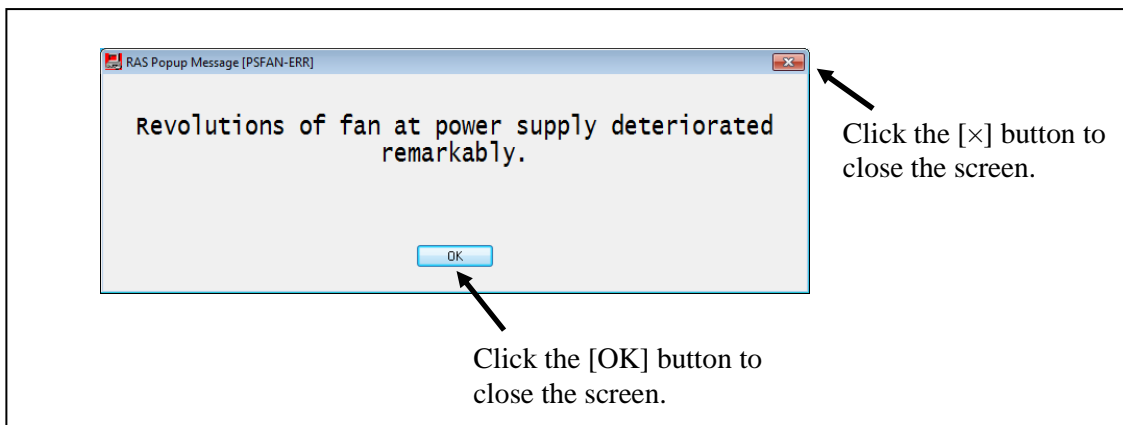


Figure 4-9 Example of Popup Message Display

---

#### NOTE

This feature is not supported for Model 30.

---

### 4.3.2 Contents of displayed messages

The following shows the message contents displayed by this feature.

Note that the displayed message contents can be edited. For details about how to editing them, see “3.1.4 Editing popup notification messages.”

Table 4-8 Displayed Message Contents

No.	Event	Displayed message contents
1	PS fan failure	Revolutions of fan at power supply deteriorated remarkably.
2	CPU fan failure	Revolutions of cpu fan deteriorated remarkably.
3	Internal cubicle temperature error	Temperature exceeded prescribed value.
4	Drive failure is predicted	A failure may be imminent on the drive of the drive bay%1.
5	Drive usage time exceeds the predetermined value	Used hours on the drive of the drive bay%1 exceeded prescribed value.
6	Memory error correction occurs with high frequency	In the %1, error correctings have occurred with high frequency.
7	RAID error occurs.	RAID status is DEGRADE. (Drive bay%1)

No. 4, 5, and 7: %1 indicates the drive bay number.

No. 4: This includes cases in which drive failure prediction information could not be obtained.

No. 6: %1 indicates the DIMM name.

### 4.3.3 Popup notification feature settings

Whether this feature is used can be set in the RAS feature settings window. It is disabled in the factory default settings. When it is disabled, popup messages are not displayed.

For details, see “3.1.3 How to use the RAS feature settings window.”

## 4. CHECKING HARDWARE STATUS

### 4.4 Digital LED Feature for Status Indication [Model 35 Only]

#### 4.4.1 Overview

This feature notifies users of hardware errors and other events that should be reported to users, by displaying a status code in the digital LED for status indication mounted on the front of the device. This enables users to know when hardware errors and other events occur.

It is possible to display an arbitrary status code from user applications by using a library function. The LED display shows a 2-digit hexadecimal.

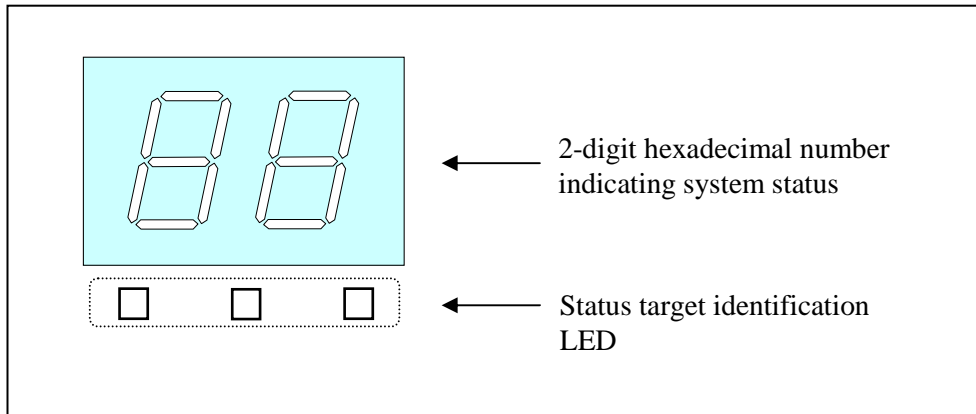


Figure 4-10 Digital LED for Status Indication

---

#### NOTE

This feature is not supported for Model 30.

---

## 4.4.2 Displayed status codes

## (1) Hardware status codes

These are the status codes displayed when an error is detected in the hardware status of this device. While the hardware status is normal, nothing is displayed.

Specifically, the hardware status code is displayed in the following cases.

- Fan failure
- Internal cubicle temperature errors
- Drive failure is predicted
- RAID error is detected (RAID model only)

When a hardware status code is displayed, the leftmost status target identification LED turns on.

For details about the contents of hardware status codes, see “4.4.4 List of hardware status codes.”

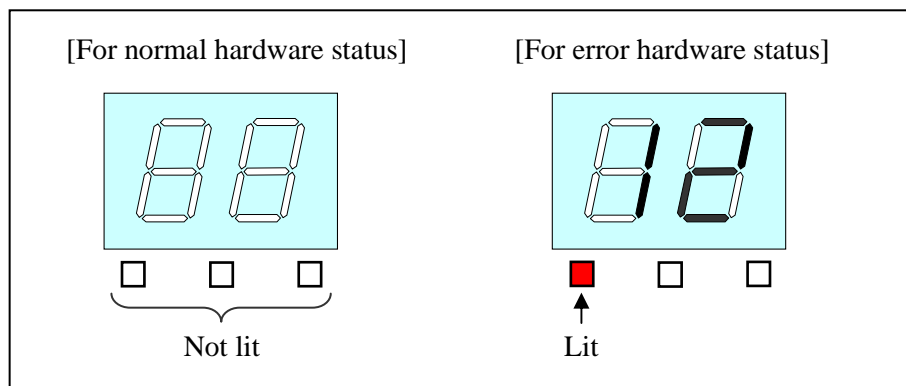


Figure 4-11 Hardware Status Codes

## (2) Application status codes

These are the status codes which are displayed by user applications using the library functions provided by this feature.

When an application status code is displayed, the middle status target identification LED turns on.

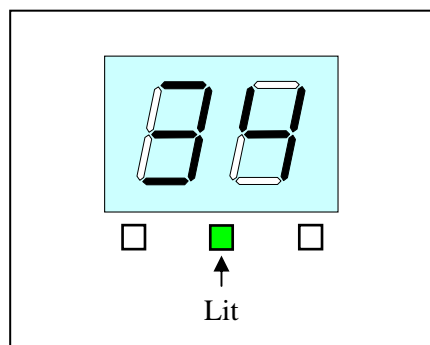


Figure 4-12 Application Status Codes

## 4. CHECKING HARDWARE STATUS

### (3) STOP error codes

These are the error codes displayed when a STOP error (blue screen) occurs due to whatever cause during system operation. “80” is given priority for display as the STOP error code, regardless of the status display mode setting. For details about the status display mode, see “4.4.3 Types of status display modes.”

When a STOP error code is displayed, all status target identification LEDs are turned on.

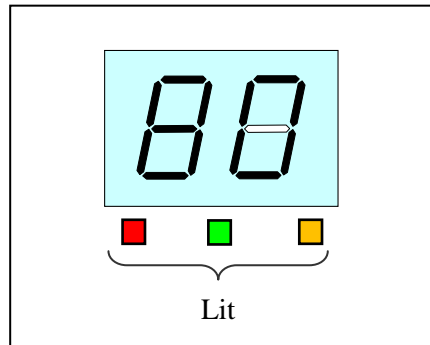


Figure 4-13 STOP Error Codes

4.4.3 Types of status display modes

This feature has two display modes: the hardware status display mode and the application status display mode.

Table 4-9 Status Display Mode

Status display mode	Description
Hardware status display mode	If the hardware status is normal, the application status code is displayed, but if an error has occurred in the hardware status, the hardware status code is given priority for display.
Application status display mode	Only application status codes are displayed. Even if an error occurred in the hardware status, the hardware status code is not displayed.

The status display mode can be set in the RAS feature settings window. The factory default setting is “hardware status display mode”.

For details about how to use the RAS feature settings window, see “3.1.3 How to use the RAS feature settings window.”

Note that if the hardware status is normal and the application status code is not specified for display, nothing is displayed on the LED.

Figure 4-14 shows an operation example of the RAS status display mode. The dotted lines in Figure 4-14 indicate the statuses for each display mode, and the bold lines show LED status transitions.

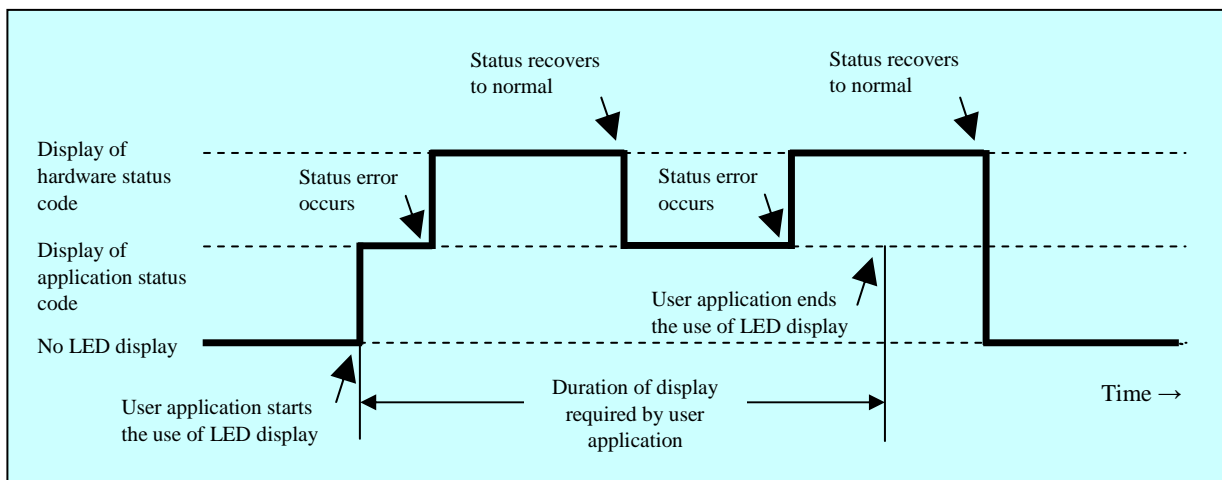


Figure 4-14 Operation Example for Status Display Modes

## 4. CHECKING HARDWARE STATUS

### 4.4.4 List of hardware status codes

Table 4-10 shows the hardware status codes that this feature displays on the digital LED for status indication.

Table 4-10 Hardware Status Code List

No.	Status code	Location for which error or failure is predicted	Priority
1	11	PS fan failure	1
2	13	CPU fan failure	
3	21	Internal cubicle temperature error	2
4	31	Predicted drive failure for drive bay 1	3
5	32	Predicted drive failure for drive bay 2	
6	41	Drive bay 1 is in offline status.	1
7	42	Drive bay 2 is in offline status.	
8	4b	RAID failure	
9	4c	RAID status unknown	
10	4d	Media error occurred in RAID	

No. 6 to 10: Displayed only for RAID model

No. 8: When RAID is in failure status, “4b” may not be displayed because the system cannot operate normally.

No. 10: Displayed only when the media error notification setting is made.

Note that if multiple failures occur at the same time, they are displayed according to priority. If two types of failure with the same priority occur, they are displayed according to the order in which errors are detected. As such, the status code of the last detected error is displayed on the digital LED for status indication.

## 4.4.5 Digital LED for status indication control function

Table 4-11 shows the functions provided as library functions to control the digital LED for status indication.

Table 4-11 List of Functions for Controlling the Digital LED for Status Indication

No.	Function name	Feature	DLL
1	SetStCode7seg	This displays the application status code.	ctrl7seg.dll
2	TurnOff7seg	This hides the application status code.	
3	SetMode7seg	This sets the status display mode.	

The above functions are provided by DLL (ctrl7seg.dll). For details about the library function interfaces, see “6.1.11 Functions for controlling the digital LED for status indication.”



### 4.5 Remote Notification Feature **[Model 35 Only]**

#### 4.5.1 Overview

This feature enables hardware statuses that could be checked only by this device to be also checked over the network from remote environments. Even in situations where this device cannot check the hardware status, such as when the system administrator is not near the device or the device is incorporated within the equipment, this feature allows hardware statuses to be checked from remote environments.

This feature uses SNMP (Simple Network Management Protocol) to report hardware statuses. This means that network management software that supports SNMP and is on the market can be used to perform centralized monitoring of multiple distributed instances of this device and other devices.

---

#### NOTE

---

- This feature is not supported for Model 30.
  - The SNMP protocol used by remote notification feature works on the TCP/IP application layer, with UDP (User Datagram Protocol) used for the transport layer. As such, depending on network load, the hardware status may not be able to be received normally.
  - Remote notification feature uses the standard Windows® SNMP service. For details about how to enable the standard Windows® SNMP service, see “4.5.3 Procedures for starting remote notification feature.”
-

## 4.5.2 Hardware statuses for which remote notification is performed

The following hardware statuses and settings can be obtained from remote environments:

- (1) Fan status
- (2) Internal cubicle temperature status
- (3) Drive status
- (4) Memory status
- (5) RAID status (RAID model only)
- (6) RAS feature settings
- (7) Operation mode (normal mode)
- (8) Version information for HF-W extension MIB (Management Information Base)

Trap notification is also performed for changes in the following hardware statuses:

- (1) Fan status
  - Normal → Error
  - Error → Normal
- (2) Internal cubicle temperature status
  - Normal → Error
  - Error → Normal
- (3) Drive status
  - Normal → Predicted failure
  - Normal → Usage time in excess of the predetermined value
- (4) Memory status
  - Normal → Frequent error correction
  - Frequent error correction → Normal
- (5) RAID status (RAID model only)
  - Normal → Error (Degraded, Unknown, Fail)
  - Error (Degraded, Unknown, Fail) → Normal
- (6) Operation mode
  - HF-W stopped → Startup in normal mode

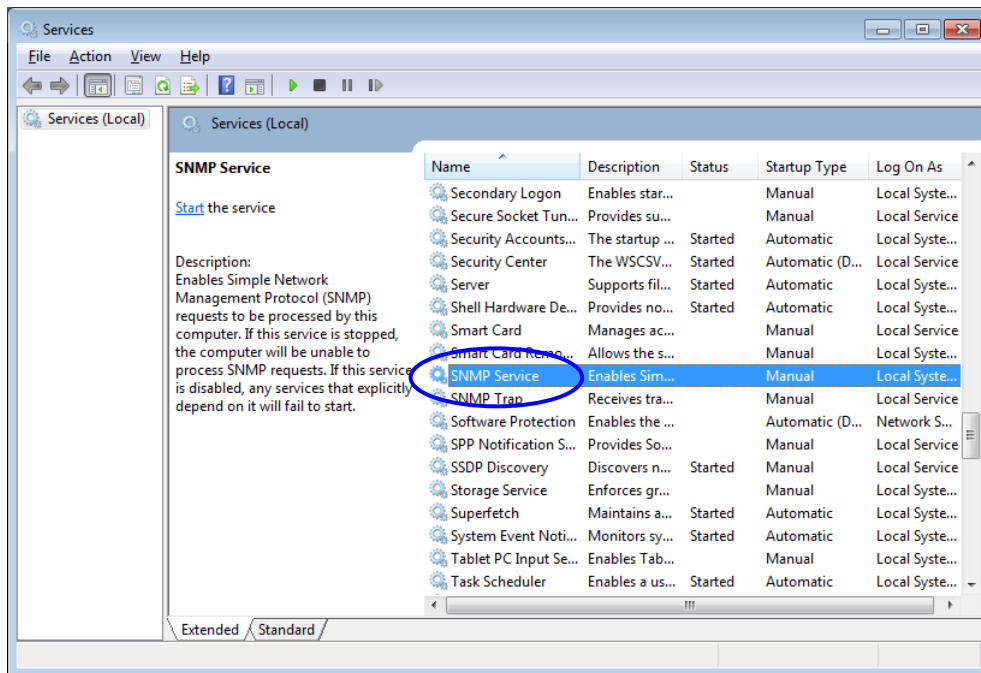
## 4. CHECKING HARDWARE STATUS

### 4.5.3 Procedures for starting remote notification feature

This feature is disabled under the factory default settings. Remote notification feature uses the standard Windows® SNMP service, and can be enabled by enabling SNMP service. To use remote notification feature, perform the following procedures to enable the SNMP service:

#### (1) Open the SNMP service properties

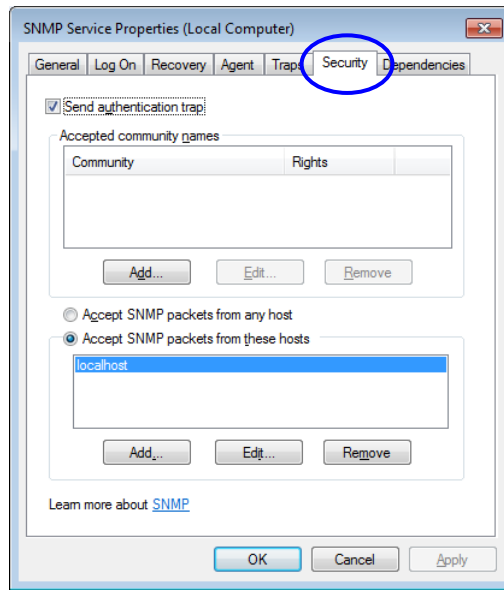
1. Log on as a computer administrator if not already logged on as such.
2. Perform the following to open the [Services] window.
  - For Windows® XP:
    - Click [Start], and then choose [Control Panel].
    - Click [Performance and Maintenance], then [Administrative Tools], and then double-click [Services].
  - For Windows® 7:
    - Click [Start], and then choose [Control Panel].
    - Click [System and Security], then [Administrative Tools], and then double-click [Services].
    - Click the [Yes] button if the [User Account Control] window is displayed.



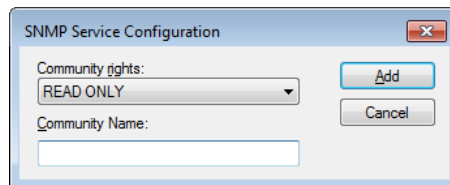
3. Double-click “SNMP Service” to display the properties window.

(2) SNMP security configuration

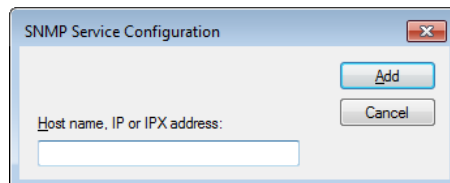
1. In the [SNMP Service Properties] window, select the [Security] tab.



2. To always send trap messages when authentication fails, select the [Send authentication trap] checkbox.
3. In “Accepted community names”, click the [Add] button to display the [SNMP Service Configuration] window. Set “READ ONLY” for “Community rights”, enter any community name for “Community Name”, and click the [Add] button.



4. Set whether to accept SNMP packets from hosts.
  - To accept SNMP packets from any manager on the network:
    - Select “Accept SNMP packets from any host”.
  - To restrict SNMP packets:
    - Select “Accept SNMP packets from these hosts”.
    - Click the [Add] button.
    - The [SNMP Service Configuration] window appears. Enter “Host name, IP or IPX address” for the restriction target, and click the [Add] button.

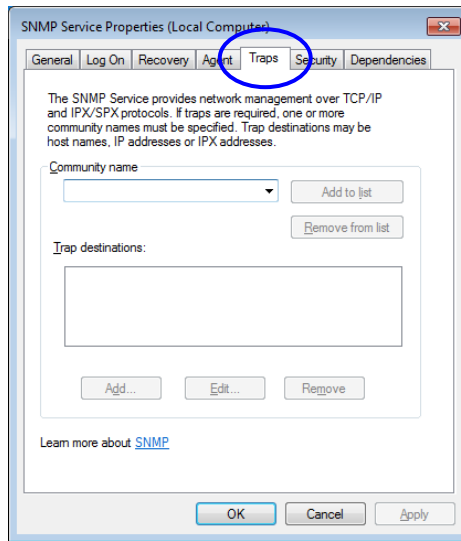


5. In the [SNMP Service Properties] window, click the [Apply] button.

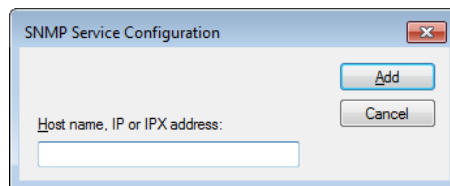
## 4. CHECKING HARDWARE STATUS

### (3) SNMP trap configuration

1. In the [SNMP Service Properties] window, select the [Traps] tab.



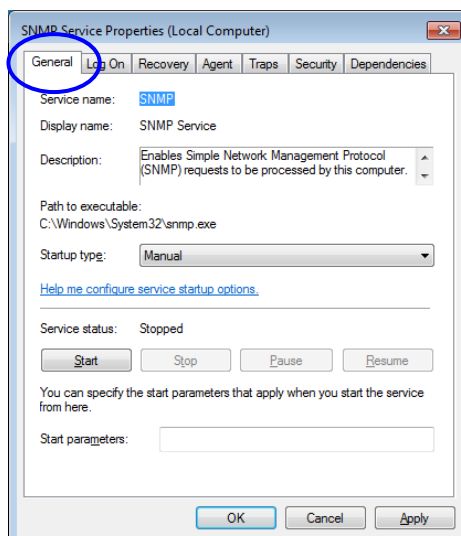
2. Enter the destination to which to send the trap messages in “Community name”, and click the [Add to list] button.
3. Click the [Add] button in “Trap destinations” to display the [SNMP Service Configuration] window. Enter “Host name, IP or IPX address” of the trap destination, and click the [Add] button.



4. In the [SNMP Service Properties] window, click the [Apply] button.

### (4) Start the SNMP service

1. In the [SNMP Service Properties] window, select the [General] tab.



2. Click the [Start] button. The SNMP service is started, and hardware status remote notification feature is enabled.
3. To enable the SNMP service to be automatically started upon next startup, set “Startup type” to “Automatic”.
4. In the [SNMP Service Properties] window, click the [OK] button.

---

#### NOTE

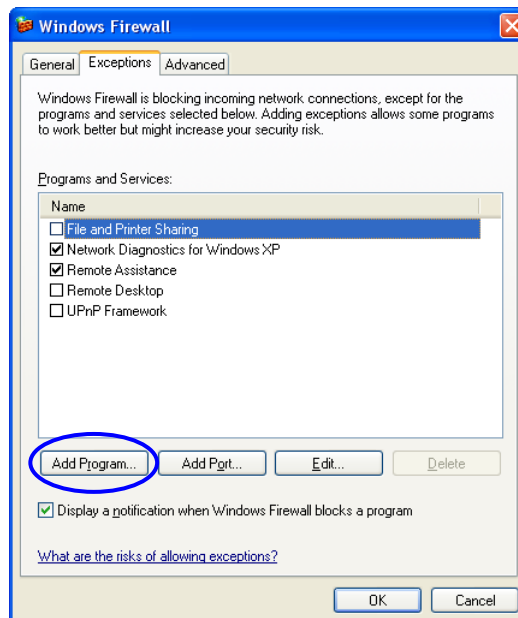
- If an error subject to trap notification has occurred before the SNMP service is started, trap notification for the error is performed with the SNMP service start timing.
- If the Windows Firewall is enabled for the SNMP service, the hardware status cannot be obtained from a remote environment.

For Windows® XP, since the Windows Firewall is enabled by default for the SNMP service, perform the following procedures to disable it.

For Windows® 7, since the Windows Firewall is disabled by default for the SNMP service, these procedures do not need to be performed. However, if the firewall is enabled, perform the following procedures to disable it.

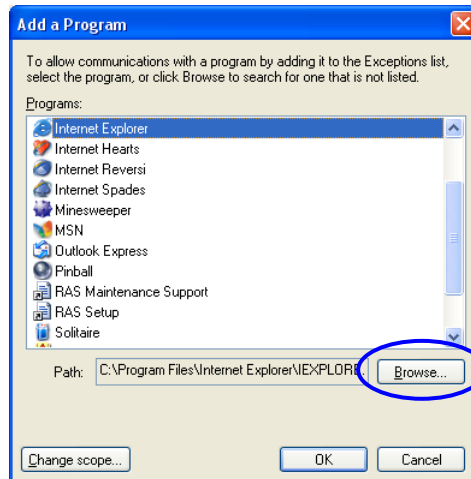
- For Windows® XP

1. Perform the following to open the [Windows Firewall] window.
  - Click [Start], and then choose [Control Panel].
  - Click [Security Center].
  - Under [Manage security settings for], click [Windows Firewall].
2. In the [Windows Firewall] window, click the [Exceptions] tab, and click the [Add Program] button.



#### 4. CHECKING HARDWARE STATUS

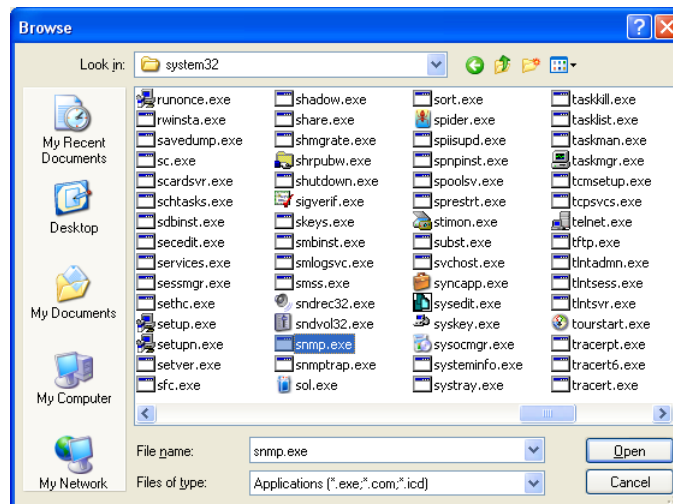
3. The [Add a Program] window appears. Click the [Browse] button.



4. Select the following program in the Browse window, and click the [Open] button.

C:\Windows\system32\snmp.exe

(Set the system drive character in the underlined area.)



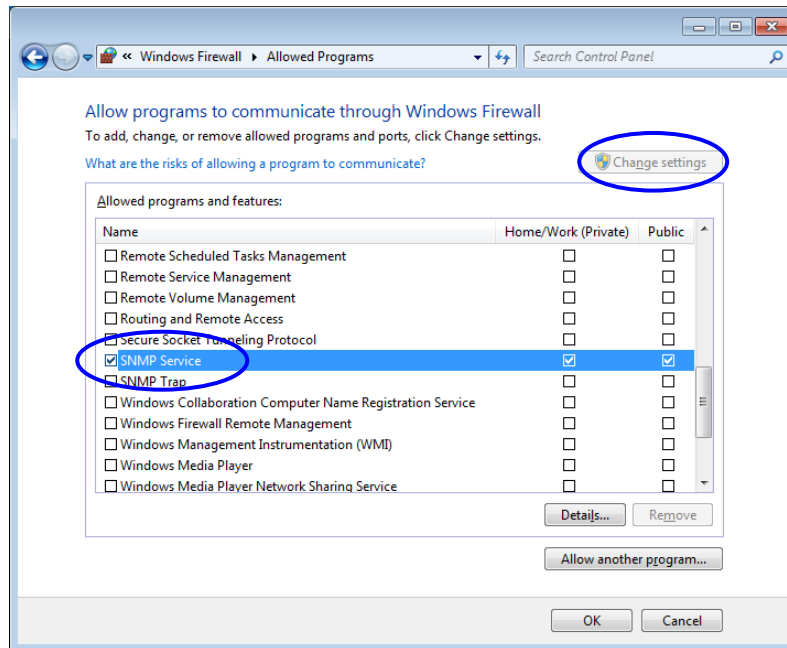
5. In the [Add a Program] window, click the [OK] button.

6. In the [Windows Firewall] window, click the [OK] button.

#### ● For Windows® 7

1. Log on as a computer administrator if not already logged on as such.
2. Click [Start], and then choose [Control Panel]. Click [System and Security].
3. In [Windows Firewall], click [Allow a program through Windows Firewall].
4. If the [User Account Control] window appears, click the [Yes] button.

5. The [Allowed Programs] window appears. Click the [Change settings] button, and in “Allowed programs and features”, select the [SNMP Service] checkbox.



6. Click the [OK] button.
-



## 4. CHECKING HARDWARE STATUS

### 4.5.4 List of extension MIB objects for HF-W

To obtain the hardware status of this device remotely, the extension MIB for HF-W is used. This section explains the objects defined by the extension MIB for HF-W.

#### (1) Objects related to hardware status and settings

Table 4-11 explains the objects related to hardware statuses. The object ID for each object is as shown below with an object or object number specified for x or y.

**Object ID : .iso.org.dod.internet.private.enterprises.Hitachi.systemExMib.  
hfwExMib.hfwRasStatus.x (where x is an object in the following table)  
or  
.1.3.6.1.4.1.116.5.45.1.y (where y is an object number in the following  
table)**

Table 4-12 Objects Related to Hardware Status

(1/2)

No.	Object	Object number	Description	Description of value
1	hfwFan	1	Fan status group	–
2	hfwFan.fanNumber	1.1	Number of fans monitored	–
3	hfwFan.fanTable.fanEntry.fanIndex	1.2.1.1	fanEntry index number	–
4	hfwFan.fanTable.fanEntry.fanName	1.2.1.2	Fan name	PS fan: Power fan CPU fan: CPU fan
5	hfwFan.fanTable.fanEntry.fanStatus	1.2.1.3	Fan status	1: Normal 2: Error
6	hfwTemp	2	Temperature status group	–
7	hfwTemp.tempNumber	2.1	Number of temperatures monitored	–
8	hfwTemp.tempTable.TempEntry.tempIndex	2.2.1.1	tempEntry index number	–
9	hfwTemp.tempTable.TempEntry.tempName	2.2.1.2	Name of monitored temperature	Internal temperature: Internal cubicle temperature
10	hfwTemp.tempTable.TempEntry.tempStatus	2.2.1.3	Temperature status	1: Normal 2: Error
11	hfwHdd	3	Drive status group	–
12	hfwHdd.hddNumber	3.1	Number of monitored hard disk drive bays	–
13	hfwHdd.hddTable.hddEntry.hddIndex	3.2.1.1	hddEntry index number	–
14	hfwHdd.hddTable.hddEntry.hddStatus	3.2.1.2	Drive status	1: Normal 2: Unmounted 3: Failure predicted 5: Usage time in excess of predetermined value 7: Offline status 8: Under rebuild 99: Unknown
15	hfwHdd.hddTable.hddEntry.hddUseTime	3.2.1.3	Drive usage time (in hours)	–
16	hfwMem	5	Memory status group	–
17	hfwMem.memNumber	5.1	Monitored memory count	–

(2/2)

No.	Object	Object number	Description	Description of value
18	hfwMem.memTable.memEntry.memIndex	5.2.1.1	memEntry index	–
19	hfwMem.memTable.memEntry.memName	5.2.1.2	Memory name	DIMM A: DIMM A slot DIMM B: DIMM B slot
20	hfwMem.memTable.memEntry.memStatus	5.2.1.3	Memory status	1: Normal 2: Error (error correction occurring frequently) 3: Unmounted
21	hfwRaid	4	RAID status group	–
22	hfwRaid.raidNumber	4.1	Number of arrays to be monitored	–
23	hfwRaid.raidTable.raidEntry.raidIndex	4.2.1.1	Index umber of raidEntry	–
24	hfwRaid.raidTable.raidEntry.raidStatus	4.2.1.2	RAID status	1: Normal 2: Degrade 4: Unknown 5: Fail
25	hfwRaid.raidTable.raidEntry.raidLevel	4.2.1.3	RAID level	1: RAID1
26	hfwRaid.raidTable.raidEntry.raidProgress	4.2.1.4	Progress in rebuilding (%)	–
27	hfwRaid.raidTable.raidEntry.raidMediaError	4.2.1.5	Presence/absence of media error	1: Media error does not exist. 2: Media error exists.

No. 12: The number of monitored drive bays is set to the number of internal drives mountable on HF-W.  
“2” is set for this device.

No. 17: The monitored memory count is set to the number of memory slots.  
“2” is set for this device.

#### 4. CHECKING HARDWARE STATUS

Table 4-13 explains the objects related to RAS feature settings. The object ID for each object is as shown below with an object or object number specified for x or y.

**Object ID : .iso.org.dod.internet.private.enterprises.Hitachi.systemExMib.  
hfwExMib.hfwRasSetting.x (where x is an object in the following table)  
or  
.1.3.6.1.4.1.116.5.45.2.y (where y is an object number in the following  
table)**

Table 4-13 Objects Related to RAS Feature Settings

No.	Object	Object number	Description	Description of value
1	hfwFanAutoShutdown	1	Automatic shutdown settings for fan failure	1: Enabled 2: Disabled
2	hfwTempAutoShutdown	2	Automatic shutdown settings for high-temperature error	1: Enabled 2: Disabled
3	hfwRemoteShutdown	3	Automatic shutdown settings for remote shutdown input	1: Enabled 2: Disabled
4	hfwSmartEnableSetting	4	Drive failure prediction (SMART monitoring) settings	1: Enabled 2: Disabled

Table 4-14 explains the objects related to operation modes. The object ID for each object is as shown below with an object or object number specified for x or y.

**Object ID : .iso.org.dod.internet.private.enterprises.Hitachi.systemExMib.  
hfwExMib.hfwRasInfo.x (where x is an object in the following table)  
or  
.1.3.6.1.4.1.116.5.45.3.y (where y is an object number in the following  
table)**

Table 4-14 Objects Related to Operation Modes

No.	Object	Object number	Description	Description of value
1	hfwRasMode	1	Operation mode	1: Normal mode

Table 4-15 explains the objects related to HF-W extension MIB versions. The object ID for each object is as shown below with an object or object number specified for x or y.

**Object ID : .iso.org.dod.internet.private.enterprises.Hitachi.system.  
hfw.hfwExMibInfo.x (where x is an object in the following table)  
or  
.1.3.6.1.4.1.116.3.45.1.y (where y is an object number in the following  
table)**

Table 4-15 Objects Related to HF-W Extension MIBs

No.	Object	Object number	Description	Description of value
1	Version	1	HF-W extension MIB version number	–
2	Revision	3	HF-W extension MIB revision number	–

#### 4. CHECKING HARDWARE STATUS

(2) Objects related to trap notifications

Table 4-16 explains the objects related to trap notifications when errors occur, as well as respective notification data. The enterprise ID for trap notifications when errors occur is as follows.

**Enterprise ID : .iso.org.dod.internet.private.enterprises.Hitachi.systemAP.  
hfwMibTrap.hfwRasErrorTrap  
or  
.1.3.6.1.4.1.116.7.45.1**

Table 4-16 Objects Related to Trap Notification (When Errors Occur)

No.	Object	Trap number	Description	Notification data	
				Target object	Value
1	hfwFanError	1	Fan failure	fanName	Name of the fan where the error occurred
				fanStatus	2: Error
				hfwFanStMsg	Revolution of %1 deteriorated remarkably.
2	hfwTempError	2	Internal cubicle temperature error	tempName	Internal temperature
				tempStatus	2: Error
				hfwTempStMsg	Internal temperature exceeded prescribed value.
3	hfwSmartDetect	3	Drive failure predicted	hddIndex	Drive bay number for which failure is predicted (SMART)
				hddStatus	3: Failure predicted
				hfwSmartStMsg	A failure may be imminent on the drive of the drive bay%2.
4	hfwHddOverRun	4	Drive usage time in excess of predetermined value	hddIndex	Drive bay number of the drive whose usage time was exceeded
				hddStatus	5: Usage time in excess of predetermined value
				hfwHddUseTimeStMsg	Used hours on the drive of the drive bay%2 exceeded prescribed value.
5	hfwMemError	6	Error correction occurring frequently	memName	One of the following: DIMM A, DIMM B
				memStatus	2: Error (Error correction occurring frequently)
				hfwMemStMsg	In the %3, error correcting have occurred with high frequency.
6	hfwArrayError	7	RAID status error occurred.	raidIndex	1
				hfwArrayStMsg	An error occurred on Array%4.

No. 1: The name of the fan in which the error has occurred is set for %1.

No. 3, 4: The number of the target drive bay is set for %2.

No. 5: The name of the memory in which error correction occurred frequently is set for %3.

No. 6: The array number where an error occurred is set for %4. "1" is always set for this device.

Table 4-17 explains the objects related to trap notification when recovery from error is performed. The enterprise ID for trap notifications during recovery from error is as follows.

**Enterprise ID : .iso.org.dod.internet.private.enterprises.Hitachi.systemAP.  
hfwMibTrap.hfwRasRecoverTrap  
or  
.1.3.6.1.4.1.116.7.45.2**

Table 4-17 Objects Related to Trap Notification (During Recovery from Error)

No.	Object	Trap number	Description	Notification data	
				Target object	Value
1	hfwFanRecover	1	Fan failure recovery	fanName	Name of the fan for which recovery was completed successfully
				fanStatus	1: Normal
				hfwFanStMsg	Revolutions of %1 returned to normal value.
2	hfwTempRecover	2	Internal cubicle temperature error recovery	tempName	Internal temperature
				tempStatus	1: Normal
				hfwTempStMsg	Internal temperature returned to prescribed value.
3	hfwMemRecover	6	Recovery from frequent error correction	memName	Name of the memory for which recovery was completed successfully
				memStatus	1: Normal
				hfwMemStMsg	In the %2, frequency of the error correctings deteriorated.
4	hfwArrayRecover	7	RAID status error recovered.	raidIndex	1
				hfwArrayStMsg	Array%3 is restored.

No. 1: The name of the successfully recovered fan is set for %1.

No. 3: The name of the successfully recovered memory is set for %1.

No. 4: The array number recovered from the error is set for %3. "1" is always set for this device.

Table 4-18 explains the objects related to trap notifications for startup in the normal mode. The enterprise ID for trap notifications related to operation modes is as follows.

**Enterprise ID : .iso.org.dod.internet.private.enterprises.Hitachi.systemAP.  
hfwMibTrap.hfwRasInfoTrap  
or  
.1.3.6.1.4.1.116.7.45.3**

Table 4-18 Objects Related to Trap Notification (Operation Mode)

No.	Object	Trap number	Description	Notification data	
				Target object	Value
1	hfwRasService Started	1	Startup in normal mode	hfwRasMode	1: Normal mode
				hfwRasStartMsg	RAS Service is running.

## 4. CHECKING HARDWARE STATUS

### 4.5.5 HF-W extension MIB file

The HF-W extension MIB file is as follows.

**HF-W extension MIB file: %ProgramFiles%\HFWRAS\mib\hfwExMib.mib**

#### 4.6 Status Acquisition Using the RAS Library **[Model 35 Only]**

The RAS library functions can be used to obtain the following hardware status. For details about library functions, see “6.1 RAS Library Interface [Model 35 Only].”

- Memory status acquisition: GetMemStatus function
- Drive status acquisition: hfwDiskStat function
- Drive failure prediction information acquisition: GetHddPredict function
- RAID status acquisition: hfwRaidStat function, RaidStat function



**This Page Intentionally Left Blank**

## CHAPTER 5 CONTROLLING HARDWARE

This device shuts down automatically upon detection of a hardware error or an incoming remote shutdown signal.

RAS library functions can also be used to control hardware status.

(1) Automatic device shutdown

This device shuts down automatically upon detection of a hardware error or an incoming remote shutdown signal. For details, see “5.1 Automatic Device Shutdown.”

(2) Controlling hardware by RAS library

The RAS library can be used from user applications to control hardware of this device. For details, see “5.2 Control by RAS Library [Model 35 Only].”

(3) Controlling RAID status by the RAID configuration control command

This command shows the status of RAID configuration drives. Also it allows you to disconnect the drive of this device that configures RAID and bring it offline. See “5.3 RAID Configuration Control Command (raidctrl) [RAID Model Only].”

## 5.1 Automatic Device Shutdown

This feature performs shutdown automatically when the device is running in a dangerous state, such as due to fan failure and high-temperature errors. This protects the processor and other built-in parts from degrading due to heat, and prevents runaway system operation due to malfunction of this device. Automatic shutdown can also be performed when a remote shutdown signal is received externally.

### 5.1.1 Automatic shutdown due to fan failure detection

When an error occurs at any fan mounted on this device, automatic shutdown is performed.

- This feature can be enabled/disabled in the RAS feature settings window. It is enabled in the factory default settings. For details, see “3.1.3 How to use the RAS feature settings window.”
- If shutdown is performed using this feature, the power is automatically cut after shutdown.
- RAS events reporting fan failure detection can also be used to perform shutdown from a user application. For details about RAS events, see “4.2 RAS Event Notification Feature [Model 35 Only].”

---

#### NOTE

---

- When use of this device is continued with a failed fan, the processor and other built-in parts may not be cooled sufficiently, causing runaway system processing and parts damage due to device malfunction. As such, leave this feature enabled whenever possible.
  - Even when not using this feature, use RAS events for reporting fan failure detection to perform shutdown from user applications.
  - For details about fan replacement procedures, refer to the *HF-W2000 Model 35/30 INSTRUCTION MANUAL* (manual number WIN-62-0060).
-

### 5.1.2 Automatic shutdown due to high-temperature error detection

Automatic shutdown is performed when the temperature sensor within this device detects that the internal cubicle temperature is too high.

- This feature can be enabled/disabled in the RAS feature settings window. It is disabled in the factory default settings. For details, see “3.1.3 How to use the RAS feature settings window.”
- If shutdown is performed using this feature, the power is automatically cut after shutdown.
- RAS events reporting high-temperature error detection can also be used to perform shutdown from a user application. For details about RAS events, see “4.2 RAS Event Notification Feature [Model 35 Only].”

---

#### NOTE

- When the internal cubicle temperature is high, since heat may cause undue parts damage, continuing to run this device is best avoided in the interests of device lifespan. However, since high-temperature errors without fan failure may indicate external causes such as an air conditioner failure where the device is located, it may be possible to resolve the cause of the high-temperature error without interrupting operation. As such, it is disabled in the factory default settings.
  - If operation of this device continues even after a high-temperature error has occurred, and if the temperature is high enough to cause dangerous internal cubicle temperatures, forced shutdown is performed and power is cut regardless of how this feature is set, to prevent system runaway and parts damage.
- 

### 5.1.3 Automatic shutdown due to remote shutdown signal detection **[Model 35 only]**

When the remote shutdown contact (RMTSTDN contact) of the RAS external control interface of this device is closed, automatic shutdown is performed. This feature allows this device to be shut down remotely from another location.

- This feature can be enabled/disabled in the RAS feature settings window. It is enabled in the factory default settings. For details, see “3.1.3 How to use the RAS feature settings window.”
- If shutdown is performed using this feature, the power is automatically cut after shutdown.
- RAS events for reporting remote shutdown signal detection can also be used to perform shutdown from a user application. For details about RAS events, see “4.2 RAS Event Notification Feature [Model 35 Only].”

---

#### NOTE

This feature is not supported for Model 30.

---

### 5.2 Control by RAS Library **[Model 35 Only]**

RAS library functions can be used to perform system shutdown, and control general external control and the digital LED for status indication. For details about library functions, see “6.1 RAS Library Interface [Model 35 Only].”

- System shutdown: BSSysShut function
- Watchdog timer control: WdtControl function
- General external control output control: GendoControlEx function and GendoControl function
- General external control input control: GetGendiEx function and GetGendi function
- Digital LED for status indication control: SetStCode7seg function, TurnOff7seg function, and SetMode7seg function

### 5.3 RAID Configuration Control Command (raidctrl) [RAID Model Only]

The raidctrl command is used to show the status of drives which configure the RAID. It also enables you to disconnect a drive forcibly or to change the media error notification setting. You can execute this command from the command prompt. Usage of this command is described below.

<Name>

raidctrl - Showing the status of RAID configuration drives and controlling the configuration

<Format>

raidctrl [/OFFLINE HDDNO] [/NOTIFY {ON|OFF} ]

<Description>

The raidctrl command enables you to show the statuses of RAID and RAID configuration drives, to control the configuration, and to change media error notification setting. This command provides the following functions depending on the specified option.

- No option specified: Shows the statuses of RAID and RAID configuration drives.
- /OFFLINE option specified: Disconnects the specified drive forcibly to bring it offline.
- /NOTIFY option specified: Specifies whether to notify a media error (when it occurs) via the interface of this device.

---

#### NOTE

- The /OFFLINE option to disconnect a drive is not supported for Model 30.
  - The following operations require the administrator privilege. Log on to the computer using the administrator account. When the user account control (UAC) is enabled in Windows® 7, open the command prompt as the administrator according to the procedure described in “APPENDIX A RUNNING THE COMMAND PROMPT AS AN ADMINISTRATOR” before executing the raidctrl command.
    - (a) Disconnection of a drive by the /OFFLINE option
    - (b) Change of media error notification setting by the /NOTIFY option
 These operations are unavailable in the remote session.
  - Do not execute more than one raidctrl command at a time.
-

## 5. CONTROLLING HARDWARE

### (1) Showing the RAID and drive statuses (no option specified)

When you execute the raidctrl command with no option specified, the RAID status and the RAID configuration drive status are displayed. The displayed RAID and RAID configuration drive statuses are described in Table 5-1 and Table 5-2 below.

Table 5-1 RAID Status Displayed by the raidctrl Command

No.	Status	Meaning
1	OPTIMAL	Normal operation.
2	DEGRADE	Degraded operation.
3	DEGRADE (REBUILD xx%)	RAID is under rebuild.
4	FAIL	Failure.
5	UNKNOWN	Status is unknown.

No. 1 to 3: "MEDIA ERROR" is added when a media error occurred.

Example: When the RAID system is normally running even though a media error occurred, the status is displayed as:

OPTIMAL (MEDIA ERROR)

No. 3, 4: Displayed only for Model 35

Table 5-2 RAID Configuration Drive Status Displayed by the raidctrl Command

No.	Status	Meaning
1	ONLINE	Normal operation.
2	OFFLINE	The drive is disconnected from RAID.
3	OFFLINE (INCOMPLETE DATA)	The drive is disconnected from RAID because of data mismatch between drives.
4	REBUILD	RAID is under rebuild (being copied).
5	NOT CONNECTED	Not connected.
6	UNKNOWN	Status is unknown.

No. 3: Displayed only for Model 30

No. 4 and 5: Displayed only for Model 35.

An example is shown below for the raidctrl command execution with no option specified.

```
C:\>raidctrl
ARRAY STATUS
 1    DEGRADE
HDD STATUS
 1    ONLINE
 2    OFFLINE
C:\>
```

## (2) Disconnecting a drive (/OFFLINE option specified) [Mode 35 only]

When you execute the `raidctrl` command with the `/OFFLINE` option specified, the specified drive is disconnected forcibly and becomes the offline mode. Use of this option requires the administrator privilege. Log on to the computer using the administrator account. If you do not have the administrator privilege, an error message appears and the process exits.

When the user account control (UAC) is enabled in Windows® 7, open the command prompt as the administrator according to the procedure described in “APPENDIX A RUNNING THE COMMAND PROMPT AS AN ADMINISTRATOR” before executing the `raidctrl` command. This option is available only when the RAID status is “OPTIMAL”. The value of `HDDNO` must be 1 or 2. If any other value is specified, an error message appears and the process exits.

Execute the following command to disconnect a RAID drive.

- To disconnect the drive bay 1 (`HDDNO=1`): `raidctrl /OFFLINE 1`
- To disconnect the drive bay 2 (`HDDNO=2`): `raidctrl /OFFLINE 2`

An example is shown below for the `raidctrl` command execution to disconnect a drive bay. A confirmation message appears to ask whether to execute disconnection. Press the `[y]` key to execute disconnection. Pressing the `[n]` key cancels the operation and terminates the command.

```
C:\ >raidctrl /OFFLINE 1
The drive of the drive bay1 will be made offline.
Are you sure you want to continue with this process? [y/n] y
An offline request on the drive of the drive bay1 was sent.
C:\ >
```

---

**NOTE**

---

The disconnected drive cannot be used as an online drive again.  
Before disconnecting a drive, make sure that you are going to disconnect the correct one.

---



## 5. CONTROLLING HARDWARE

### (3) Changing the setting of whether a media error occurrence is to be notified or not (/NOTIFY option specified)

When you execute the `raidctrl` command with the `/NOTIFY` option specified, the setting of whether a media error occurrence should be notified can be changed.

To change the media error notification setting by specifying `ON` or `OFF` in this option, the administrator privilege is required. Log on to the computer using the administrator account. If you do not have the administrator privilege, an error message appears and the process exits.

When the user account control (UAC) is enabled in Windows® 7, open the command prompt as the administrator according to the procedure described in “APPENDIX A RUNNING THE COMMAND PROMPT AS AN ADMINISTRATOR” before executing the `raidctrl` command. When only the `/NOTIFY` option is specified, the current notification setting is shown.

An example is shown below for the `raidctrl` command execution with the `/NOTIFY` option specified.

- To confirm the media error notification setting: `raidctrl /NOTIFY`

```
C:\ >raidctrl /NOTIFY
MEDIA ERROR Notify Setting:ON
C:\ >
```

- To enable the media error notification: `raidctrl /NOTIFY ON`
- To disable the media error notification: `raidctrl /NOTIFY OFF`

## &lt;Diagnosis&gt;

When the process terminates successfully, the raidctrl command returns the exit code 0.

When the process terminates unsuccessfully, the raidctrl command shows an error message of Table 5-3 below and returns an exit code other than 0.

For the error messages No. 4 and No. 5, however, it returns the exit code 0.

Table 5-3 Error Messages of the raidctrl Command

No.	Error message	Meaning
1	Usage:raidctrl [/OFFLINE HDDNO] [/NOTIFY [{ON OFF}]]	An invalid option is specified. Specify the correct option.
2	Invalid argument. (%1)	The specified option %1 is invalid.
3	Specified HDD is not valid.	The specified drive is not connected, or it's status is unknown.
4	Specified HDD is already OFFLINE.	The specified drive has already been disconnected.
5	Specified HDD is the only ONLINE drive.	The specified drive is the last normal drive.
6	An error occurred in %1. errorcode = %2	An unexpected error (%2) occurred in the function (%1). Execute the command again. If this error message appears again, restart this device.
7	Access denied. Log on as any user that is a member of the Administrators group, and execute this command again.	You do not have the administrator privilege. Log on as a member of the Administrator group and execute the command again. [For Windows XP]
8	You do not have the privilege to run this command option. Please run this command option again on "Administrator: Command Prompt".	You do not have the administrator privilege. Log on to the computer using the administrator account and execute the command again. If the user account control (UAC) is enabled, execute the command on the "Administrator: Command Prompt" screen. [For Windows 7]
9	Access denied. Log on console session, And execute this command again.	You cannot use an option of this command from the remote session. Log on to the console session and execute the command again. [For Windows XP]
10	Access denied. Log on console session, And execute this command again on "Administrator: Command Prompt".	You cannot use an option of this command from the remote session. Log on to the console session and execute the command again. If the user account control (UAC) is enabled, execute the command on the "Administrator: Command Prompt" screen. [For Windows 7]
11	Specified HDD is rebuilding.	The specified drive is under rebuild.
12	/OFFLINE option is not supported.	The /OFFLINE option is not supported.

No. 3 to 5, 11: Displayed only for Model 35

No. 12: Displayed only for Model 30

**This Page Intentionally Left Blank**

## CHAPTER 6 LIBRARY FUNCTIONS

The RAS library can be used from user applications, to obtain and control the hardware status of this device.

For details about the hardware specifications for the RAS external control interface (options) and meaning of each contact described in this chapter, refer to the *HF-W2000 Model 35/30 INSTRUCTION MANUAL* (manual number WIN-62-0060).

### 6.1 RAS Library Interface [Model 35 Only]

#### 6.1.1 Overview

This chapter describes the interfaces for the functions provided by the RAS library.

Table 6-1 lists the functions provided by the RAS library.

Table 6-1 List of Functions Provided by the RAS Library

No.	Function name	Feature	DLL
1	BSSysShut	Performs shutdown.	w2kras.dll
2	WdtControl	Retrigger, obtains the status of, and stops the watchdog timer.	
3	GendoControl	Opens and closes the general external control output (GENDO0). (for backward compatibility)	
4	GendoControlEx	Opens and closes the general external control output (GENDO0, GENDO1, GENDO2).	
5	GetGendi	Obtains the status of the general external control input (GENDI). (for backward compatibility)	
6	GetGendiEx	Obtains the status of the general external control input (GENDI, GENDI0, GENDI1, GENDI2).	
7	MconWriteMessage	Logs any string to a log file specific to this device.	
8	GetMemStatus	Obtains the status of memory mounted on this device.	
9	GetHddPredict	Obtains information about drive failure prediction. (for backward compatibility)	
10	hfwDiskStat	Obtains the drive status.	hfwras.dll
11	hfwRaidStat	Obtains the RAID status.	hfwraid.dll
12	RaidStat	Obtains the status of RAID configuration drives. (For downward compatibility)	
13	SetStCode7seg	Displays the application status code on the digital LED for status indication.	ctrl7seg.dll
14	TurnOff7seg	Hides the application status code for the digital LED for status indication.	
15	SetMode7seg	Sets the status display mode for the digital LED for status indication.	

The above functions are provided as DLLs (w2kras.dll, hfwras.dll, hfwraid.dll, and ctrl7seg.dll).

---

NOTE

---

- This feature is not supported for Model 30.
- In Windows® 7 (64-bit), those DLLs can be linked into only 64-bit applications. 32-bit applications cannot use those DLLs.

The following function has been provided to maintain backward compatibility with the Windows NT® programs and old equipment, but Windows® 7 (64-bit) does not support this function. Use the function provided as an alternative.

Function not supported in Windows® 7 (64-bit)	Alternative function
GetHddPredict	hfwDiskStat
RaidStat	hfwRaidStat

- Do not copy or move w2kras.dll, hfwras.dll, hfwrraid.dll, and ctrl7seg.dll to another directory, as this may prevent RAS feature from operating normally for this device.
-

The functions provided in w2kras.dll and ctrl7seg.dll can also be called from Visual Basic® (.NET is required for Windows® 7(64-bit)). When calling No. 1 to No. 8, or No. 11 to No. 13 from Visual Basic®, use the above function name with `_VB` appended to the end. The function parameters are the same. For example, to call the `WdtControl` function from Visual Basic®, use the `WdtControl_VB` function name.

The following import libraries are provided:

```
%ProgramFiles%\HFWRAS\lib\w2kras.lib
%ProgramFiles%\HFWRAS\lib\hfwras.lib
%ProgramFiles%\HFWRAS\lib\ctrl7seg.lib
%ProgramFiles%\HFWRAS\lib\hfwraid.lib
```

When using these libraries, link the import library corresponding to each library.

The following header files are provided for this library:

```
%ProgramFiles%\HFWRAS\include\w2kras.h
%ProgramFiles%\HFWRAS\include\hfwras.h
%ProgramFiles%\HFWRAS\include\ctrl7seg.h
%ProgramFiles%\HFWRAS\include\hfwraid.h
```

When using them in C, include the header file corresponding to each library.

<Migrating from Windows® XP applications>

- If the application uses only the functions listed in Table 6-1, object compatibility is maintained.
- In Windows® 7 (32-bit), you can execute the application as is.
- In Windows® 7 (64-bit), you need to link the libraries from a 64-bit version of the application.

<Migrating from Windows® 2000 applications>

- If the application uses only the functions Nos.1 through 8 listed in Table 6-1, object compatibility is maintained.
- In Windows® 7 (32-bit), you can execute the application as is. It should be noted that due to the specifications of the OSs, the behavior of the `BSSysShut` function is different from when the function is executed on Windows® 2000. When you specify 0 for the first argument, an “It is now safe to turn off your computer” message box is not displayed, and the power is turned off after the shutdown. Also note that the `BSSysShutEx` function that has been provided for Windows® 2000 is no longer available.
- In Windows® 7 (64-bit), you need to link the libraries from a 64-bit version of the application.

## 6. LIBRARY FUNCTIONS

### 6.1.2 Shutdown function (BSSysShut)

#### <Name>

BSSysShut - System shutdown

#### <Format>

```
#include <w2kras.h>
int BSSysShut(reboot)
int reboot: /* Reboot specification flag */
```

#### <Description>

BSSysShut performs system shutdown processing.

The reboot argument specifies whether to restart the system after shutdown.

reboot=0: This device will be powered off after shutdown.

reboot ≠ 0: The system will restart after shutdown.

#### <Diagnostics>

0: Normal termination (system shutdown processing was started)

1: Error in obtaining shutdown permissions

2: Internal error (OS shutdown failure)

#### <Sample program>

A sample program in C is provided as a usage example of this function. For details about the storage location and file name for this sample program, see “6.1.12 Sample programs.”

## 6.1.3 Watchdog timer control function (WdtControl)

## (1) Function interface

&lt;Name&gt;

WdtControl - Controlling / Obtaining the status for the watchdog timer

&lt;Format&gt;

#include &lt;w2kras.h&gt;

BOOL WdtControl(DWORD dwCmd, PDWORD pdwCount);

&lt;Description&gt;

This function performs the processing specified by dwCmd for the watchdog timer.

When using this function, select “Retriggered by application program” under “Watchdog timer setting” in the RAS feature settings window. If any other watchdog timer setting is selected, this function will terminate with an error. In this case, the Windows API GetLastError function returns the W2KRAS\_WDT\_NONMANUAL error code when called.

The following explains the parameters of this function.

dwCmd:

Specifies the processing for the watchdog timer. The values that can be specified for this parameter are as follows.

Table 6-2 Processing Specified for dwCmd of WdtControl

dwCmd	Description
WDT_SET	Sets the timeout time (in seconds).
WDT_STOP	Stops the watchdog timer.
WDT_STAT	Obtains the status of the watchdog timer.

If any values other than those above are specified, this function terminates with an error, and the W2KRAS\_INVALID\_PARAMETER error code is returned when the Windows API GetLastError function is called.

pdwCount:

When dwCmd is WDT\_SET, the watchdog timer timeout time can be set by setting the time in the area indicated by pdwCount and calling this function.

The time can be set in seconds, from 1 to 63. If any other values are set, error termination is performed, and the W2KRAS\_INVALID\_PARAMETER error code is returned when the Windows API GetLastError function is called.

Since the content of the area indicated by pdwCount is uncertain when returned, do not access it.

When dwCmd is WDT\_STOP, pdwCount is ignored. Since the content of the area indicated by pdwCount is uncertain when returned, do not access it.



## 6. LIBRARY FUNCTIONS

When `dwCmd` is `WDT_STAT`, the remaining time (in seconds) from when this function is executed until the watchdog timer times out is stored in the area indicated by `pdwCount`, as the return value. If 0 is stored in the area indicated by `pdwCount` and returned, this means that a watchdog timer timeout has occurred.

Also, any value already set in the area indicated by `pdwCount` is ignored when this function is called.

### <Diagnostics>

If this function terminates normally when called, `TRUE` is returned. If it terminates with an error, `FALSE` is returned.

If error termination occurs, the error code can be obtained by calling the Windows API `GetLastError` function. The error codes returned by this function are as follows.

Error code (value)	Description
<code>W2KRAS_INVALID_PARAMETER</code> (0x2001)	The specified argument is incorrect.
<code>W2KRAS_WDT_NONMANUAL</code> (0x2002)	This function cannot be used, because “Retriggered by application program” is not selected for the watchdog timer setting in the RAS feature settings window.
<code>W2KRAS_NOT_INITIALIZE</code> (0x2005)	RAS software has not finished starting up.
<code>W2KRAS_INTERNAL_ERROR</code> (0x2007)	An internal error has occurred.

Any error codes other than the above are from the Windows API function used by this function. For details, see the help for the Windows API.

### <Sample program>

A sample program in C is provided as a usage example of this function. For details about the storage location and file name of this sample program, see “6.1.12 Sample programs.”

For details about how to use the watchdog timer to monitor program operation statuses, see “2.7.2 Monitoring the user program operation status.”

## (2) Operation of the WDTTO contact of the RAS external control interface

This section explains operation of the WDTTO contact of the RAS external control interface in each of the following statuses.

- Status during device startup

The WDTTO contact is closed.

- Status during OS startup

During OS startup, the WDTTO contact is closed. If the watchdog timer is retriggered by the watchdog timer automatic retrigger feature or the WdtControl function, the WDTTO contact is opened.

- Status when the WdtControl function is executed

When dwCmd is WDT\_SET:

The WDTTO contact is open. Then, if the watchdog timer is not retriggered within the set timeout time, the contact is closed. If the watchdog timer is retriggered while the WDTTO contact is closed, the WDTTO contact is opened.

When dwCmd is WDT\_STOP:

The WDTTO contact is open. In this case, since the watchdog timer stops the countdown, timeout does not occur.

- Status when automatic retrigger feature is used for the watchdog timer

The WDTTO contact is open. Since the process running this feature is run at the lowest level (idle priority), a watchdog timer timeout occurs when a process with a priority other than the idle priority occupies the CPU for a time longer than that set. In this case, the contact is closed. Then, if the CPU becomes free and a process can be run to execute this feature, the watchdog timer is retriggered again, in which case the WDTTO contact is opened.

Figure 6-1 shows an example of WDTTO contact operation when an application uses the WdtControl function to control the watchdog timer. Here, the timeout time of the watchdog timer specified by the WdtControl function is 10 seconds.



- 6.1.4 Output control functions for general external controls (GendoControl, GendoControlEx)  
Output to the general external controls of the RAS external control interface is controlled by the GendoControl function and GendoControlEx function.

(1) Function interface (GendoControl)

<Name>

GendoControl - Output control for the general external control (GENDO0)

<Format>

```
#include <w2kras.h>
BOOL GendoControl(DWORD dwCmd);
```

<Description>

This function performs the processing specified by dwCmd for the general external control (GENDO0) of the RAS external control interface. This function exists for backward compatibility.

Table 6-3 shows the processing specified for dwCmd.

Table 6-3 Processing Specified for dwCmd of the GendoControl Function

dwCmd	Description
GENDO_OPEN	Opens the general external control (GENDO0)
GENDO_CLOSE	Closes the general external control (GENDO0)

If any values other than those above are specified, this function terminates with an error, and the W2KRAS\_INVALID\_PARAMETER error code is returned when the Windows API GetLastError function is called.

<Diagnostics>

If this function terminates normally when called, TRUE is returned. If it terminates with an error, FALSE is returned.

If error termination occurs, the error code can be obtained by calling the Windows API GetLastError function. The error codes returned by this function are as follows.

Error code (value)	Description
W2KRAS_INVALID_PARAMETER (0x2001)	The specified argument is incorrect.
W2KRAS_NOT_INITIALIZE (0x2005)	RAS software has not finished starting up.
W2KRAS_INTERNAL_ERROR (0x2007)	An internal error has occurred.

Any error codes other than the above are from the Windows API function used by this function. For details, see the help for the Windows API.

## 6. LIBRARY FUNCTIONS

### (2) Function interface (GendoControlEx)

#### <Name>

GendoControlEx - Output control for the general external controls (GENDO0 /GENDO1 /GENDO2)

#### <Format>

```
#include <w2kras.h>
```

```
BOOL GendoControlEx(DWORD dwPort, DWORD dwCmd);
```

#### <Description>

This function performs the processing specified by dwCmd for the general external controls (GENDO0, GENDO1, or GENDO2) specified by dwPort of the RAS external control interface.

Table 6-4 shows the processing specified for dwPort of the RAS external control interface.

Table 6-4 Processing Specified by dwPort of the GendoControlEx Function

dwPort	Description
GENDO0_PORT	Runs the general external control (GENDO0)
GENDO1_PORT	Runs the general external control (GENDO1)
GENDO2_PORT	Runs the general external control (GENDO2)

Table 6-5 shows the processing specified by dwCmd.

Table 6-5 Processing Specified by dwCmd of the GendoControlEx Function

dwCmd	Description
GENDO_OPEN	Opens the general external control specified by dwCmd
GENDO_CLOSE	Closes the general external control specified by dwCmd

If any values other than those above are specified, this function terminates with an error, and the W2KRAS\_INVALID\_PARAMETER error code is returned when the Windows API GetLastError function is called.

## &lt;Diagnostics&gt;

If this function terminates normally when called, TRUE is returned. If it terminates with an error, FALSE is returned.

If error termination occurs, the error code can be obtained by calling the Windows API GetLastError function. The error codes returned by this function are as follows.

Error code (value)	Description
W2KRAS_INVALID_PARAMETER (0x2001)	The specified argument is incorrect.
W2KRAS_NOT_INITIALIZE (0x2005)	RAS software has not finished starting up.
W2KRAS_INTERNAL_ERROR (0x2007)	An internal error has occurred.

Any error codes other than the above are from the Windows API function used by this function. For details, see the help for the Windows API.

## &lt;Sample program&gt;

A sample program in C is provided as a usage example of this function. For details about the storage location and file name of this sample program, see “6.1.12 Sample programs”.

## (3) Operation of general external controls (GENDO0 /GENDO1 /GENDO2) of the RAS external control interface

These general external controls are open when the power is ON and OFF.

Figure 6-3 shows operation of the GENDO0 contact when the GendoControl function is used.

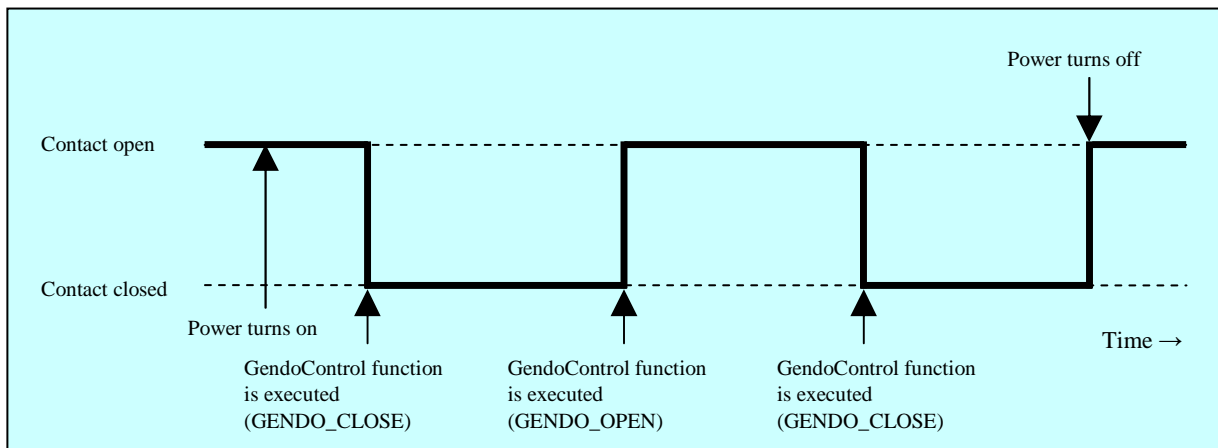


Figure 6-3 GENDO Contact Operation

The dotted lines in Figure 6-3 show each status of the corresponding contact, and the bold lines show the status transitions of the corresponding contact.

The explanations in this section are common to both RAS external control interfaces HJ-7805-21 and HJ-7805-22.

## 6. LIBRARY FUNCTIONS

### 6.1.5 Input status acquisition functions for general external controls (GetGendi, GetGendiEx)

The status of input to general external controls of the RAS external control interface is obtained by the GetGendi function and GetGendiEx function.

#### (1) Function interface (GetGendi)

##### <Name>

GetGendi - Obtaining the status of the GENDI general external control input

##### <Format>

```
#include <w2kras.h>
DWORD GetGendi(VOID);
```

##### <Description>

This function obtains the status of the general external control input (GENDI) of the RAS external control interface. This function exists for backward compatibility.

When using this function, clear the [Automatically shutdown if remote shutdown is requested] checkbox in the shutdown settings of the RAS feature settings window. If this setting is selected, this function terminates with an error, and the W2KRAS\_RMTSTDN\_ON error code is returned when the Windows API GetLastError function is called.

##### <Return values>

1: The GENDI external control is closed  
0: The GENDI external control is open  
0xffffffff: Error termination

##### <Diagnostics>

This function returns 0xffffffff when it terminates with an error upon execution.

When error termination occurs, the GetLastError function of the Windows API can be called to obtain the error code. The error codes returned by this function are as follows.

Error code (value)	Description
W2KRAS_RMTSTDN_ON (0x2003)	This function cannot be used because the [Automatically shutdown if remote shutdown is requested] checkbox is selected in the shutdown settings of the RAS feature settings window.
W2KRAS_NOT_INITIALIZE (0x2005)	RAS software has not finished starting up.
W2KRAS_INTERNAL_ERROR (0x2007)	An internal error has occurred.

Any error codes other than the above are from the Windows API function used by this function. For details, see the help for the Windows API.

##### <Sample program>

A sample program in C is provided as a usage example of this function. For details about the storage location and file name of this sample program, see “6.1.12 Sample programs.”

## (2) Function interface (GetGendiEx)

## &lt;Name&gt;

GetGendiEx - Obtaining the status of general external control input (GENDI, GENDI0, GENDI1, GENDI2)

## &lt;Format&gt;

```
#include <w2kras.h>
DWORD GetGendiEx(DWORD dwPort);
```

## &lt;Description&gt;

This function obtains the status of the general external control input (GENDI, GENDI0, GENDI1, GENDI2) of the RAS external control interface specified by dwPort.

Table 6-6 shows the processing specified for dwPort.

Table 6-6 Processing Specified by dwPort of the GetGendiEx Function

dwPort	Description
GENDI_PORT	Obtains the status of the general external control (GENDI)
GENDI0_PORT	Obtains the status of the general external control (GENDI0)
GENDI1_PORT	Obtains the status of the general external control (GENDI1)
GENDI2_PORT	Obtains the status of the general external control (GENDI2)

If any values other than those above are specified, this function terminates with an error, and the W2KRAS\_INVALID\_PARAMETER error code is returned when the Windows API GetLastError function is called.

When specifying GENDI\_PORT for dwPort to obtain the GENDI status, clear the [Automatically shutdown if remote shutdown is requested] checkbox in the shutdown settings of the RAS feature settings window. If this setting is selected, this function terminates with an error, and the W2KRAS\_RMTSTDN\_ON error code is returned when the Windows API GetLastError function is called.

## &lt;Return values&gt;

1: The external control specified by dwPort is closed  
 0: The external control specified by dwPort is open  
 0xffffffff: Error termination



## 6. LIBRARY FUNCTIONS

### <Diagnostics>

This function returns 0xffffffff when it terminates with an error upon execution.

When error termination occurs, the GetLastError function of the Windows API can be called to obtain the error code. The error codes returned by this function are as follows.

Error code (value)	Description
W2KRAS_INVALID_PARAMETER (0x2001)	The specified argument is incorrect.
W2KRAS_RMTSTDN_ON (0x2003)	The GENDI status cannot be obtained, because the [Automatically shutdown if remote shutdown is requested] checkbox is selected in the shutdown settings of the RAS feature settings window.
W2KRAS_NOT_INITIALIZE (0x2005)	RAS software has not finished starting up.
W2KRAS_INTERNAL_ERROR (0x2007)	An internal error has occurred.

Any error codes other than the above are from the Windows API function used by this function. For details, see the help for the Windows API.

### <Sample program>

A sample program in C is provided as a usage example of this function. For details about the storage location and file name of this sample program, see “6.1.12 Sample programs.”

## 6.1.6 Log information recording function (MConWriteMessage)

## &lt;Name&gt;

MConWriteMessage - Recording log information

## &lt;Format&gt;

```
#include <w2kras.h>
VOID WINAPI MConWriteMessage(LPSTR lpBuffer);
```

## &lt;Description&gt;

The MConWriteMessage function writes the specified string data to a log file (hfwrasa.log or hfwrasb.log).

The string data is recorded along with the log collection time.

There are two log files, with the size of each being 64 KB. If the logs recorded to the currently used log file exceed 64 KB, the recorded log file is switched to another log file.

The following explains the parameters of this function.

## lpBuffer:

Specifies a pointer to the string of data to be written.

Set the application name or other information at the start of the specified message to facilitate identification of the application recording the log information.

## &lt;Viewing log information&gt;

The log information recorded by this function is stored as text in the following files. If the logs recorded to the currently used log file exceed 64 KB, the recorded log file is switched to another log file.

- %ProgramFiles%\HFWRAS\log\hfwrasa.log
- %ProgramFiles%\HFWRAS\log\hfwrasb.log

The above files can be opened in an application such as Notepad to view log information. The format for the log information is as follows.

<pre>YYYY/MM/DD hh:mm:ss - &lt;specified-log-information&gt; YYYY/MM/DD hh:mm:ss - &lt;specified-log-information&gt; YYYY/MM/DD hh:mm:ss - &lt;specified-log-information&gt; :</pre>	<pre>YYYY: Year MM: Month DD: Day hh: Hours (24-hour display) mm: Minutes ss: Seconds</pre>
--	---

Figure 6-4 Log Information Format

Note that in the initial status of the above files, an EOF (ASCII code: 0x1a) is set according to the size of the file.

## &lt;Sample program&gt;

A sample program in C is provided as a usage example of this function. For details about the storage location and file name of this sample program, see “6.1.12 Sample programs.”

---

### NOTE

---

- This function has the same name as the message console output function provided in our W2K-PLUS software, but does not perform output to the message console.
  - This function terminates asynchronously, without waiting for the data to be actually written to the log file. As such, no error is returned if attempts to write to the log file failed for whatever reason. We recommend keeping important information in the OS event log.
  - To minimize resource usage, this function performs pipe open/close processing when called. As such, since this function may have relatively significant processing overhead, call it once and output multiple lines of logs at a time whenever possible.
  - This function does not support Unicode strings. Make sure that ANSI strings are used. The message logs are stored in a text file, in which the two characters “\r\n” are recognized as a line break.  
When using the string specified by lpbuffer, enter “\r\n” to specify a line break.
-

## 6.1.7 Memory status acquisition function (GetMemStatus)

## &lt;Name&gt;

GetMemStatus - Obtaining the memory status

## &lt;Format&gt;

```
#include <w2kras.h>
BOOL GetMemStatus(PMEM_DATA pMemData);
```

## &lt;Description&gt;

The GetMemStatus function stores the status of the memory mounted on this device, in the area specified by pMemData. The following explains the parameters of this function.

## pMemData:

This specifies a pointer to the following MEM\_DATA struct for storing the obtained memory status.

```
typedef struct MEM_DATA {
    int          Dimm_Number; // DIMM slot count in the device
    DWORD       Dimm_Status[4]; // Each DIMM status
} MEM_DATA, *PMEM_DATA;
```

If this function is successful, the Dimm\_Number is set to the number of DIMMs that can be mounted on the device. This value differs depending on hardware type. The values shown in the table are set for each element of Dimm\_Status. In this case, the number of valid elements is the value returned by Dimm\_Number (For example, if the value returned by Dimm\_Number is 2, elements as far as Dimm\_Status[1] are valid). Any subsequent elements are for future use. Since the value is uncertain, do not access it.

Table 6-7 Values for Each Element of Dimm\_Status

Value	Meaning
MEMORY_NOMAL	Memory is running normally.
MEMORY_ERR_DETECT	Error correction is occurring frequently.
MEMORY_NOT_MOUNTED	Memory is not mounted.

Note that for this hardware type, the relationship between each Dimm\_Status element and the DIMM name is as follows.

Element	DIMM name
Dimm_Status[0]	DIMM A
Dimm_Status[1]	DIMM B

## 6. LIBRARY FUNCTIONS

### <Diagnostics>

If this function terminates normally when called, TRUE is returned. If it terminates with an error, FALSE is returned. In the case of error termination, the value stored in pMemData is invalid. If error termination occurs, the error code can be obtained by calling the Windows API GetLastError function. The error codes returned by this function are as follows.

Error code (value)	Description
W2KRAS_INVALID_PARAMETER (0x2001)	The specified argument is incorrect.
W2KRAS_NOT_INITIALIZE (0x2005)	RAS software has not finished starting up.
W2KRAS_MEMST_INVALID (0x2007)	The memory status cannot be obtained.

Any error codes other than the above are from the Windows API function used by this function. For details, see the help for the Windows API.

### <Sample program>

A sample program in C is provided as a usage example of this function. For details about the storage location and file name of this sample program, see “6.1.12 Sample programs.”

## 6.1.8 Function for obtaining drive failure prediction information (GetHddPredict)

## &lt;Name&gt;

GetHddPredict - Function for obtaining drive failure prediction information

## &lt;Format&gt;

```
#include <w2kras.h>
BOOL GetHddPredict(PHDD_PREDICT_DATA pHddData);
```

## &lt;Description&gt;

This function obtains failure prediction information for the internal drive of this device. When using this function, select the [Function is available] checkbox in the “Drive failure prediction setting (SMART)” of the RAS feature settings window. If this setting is cleared, this function terminates with an error. In this case, W2KRAS\_HDDPREDICT\_INVALID error code is returned when the Windows API GetLastError function is called. The following explains the parameters of this function.

---

**NOTE**


---

Windows® 7 (64-bit) does not support this function. Use the function provided as an alternative (hfwDiskStat).

---

## pHddData:

Specifies a pointer to the following HDD\_PREDICT\_DATA struct for storing the obtained drive failure prediction information.

```
typedef struct HDD_PREDICT_DATA {
    DWORD Hdd1_Predict; // Drive failure prediction information for drive bay 1
    DWORD Hdd2_Predict; // Drive failure prediction information for drive bay 2
} HDD_PREDICT_DATA, *PHDD_PREDICT_DATA;
```

If this function is successful, the values shown in Table 6-8 will be set in each element of the above HDD\_PREDICT\_DATA struct.

Table 6-8 Value for Each Element of the HDD\_PREDICT\_DATA Struct

Value	Meaning
HDD_PREDICT_OFF	Operation is normal.
HDD_PREDICT_ON	Failure is predicted.
HDD_NOT_CONNECTED	No connection is established.
HDD_PREDICT_FAILURE	An attempt to obtain failure prediction information failed.

## 6. LIBRARY FUNCTIONS

### <Diagnostics>

If this function terminates normally when called, TRUE is returned. If it terminates with an error, FALSE is returned. In the case of error termination, the error code can be obtained by calling the Windows API GetLastError function. The error codes returned by this function are as follows.

Error code (value)	Description
W2KRAS_INVALID_PARAMETER (0x2001)	The specified argument is incorrect.
W2KRAS_NOT_INITIALIZE (0x2005)	RAS software has not finished starting up.
W2KRAS_HDDPREDICT_INVALID (0x2006)	Drive failure prediction feature is not enabled. In the “Drive failure prediction setting (SMART)” of the RAS feature settings window, select the [Function is available] checkbox.
W2KRAS_INTERNAL_ERROR (0x2007)	An internal error has occurred.

Any error codes other than the above are from the Windows API function used by this function. For details, see the help for the Windows API.

## 6.1.9 Drive status acquisition function (hfwDiskStat)

## &lt;Name&gt;

hfwDiskStat - Obtaining drive status

## &lt;Format&gt;

```
#include <hfwras.h>
BOOL hfwDiskStat(PHFW_DISK_STATUS phfwDiskStatus);
```

## &lt;Description&gt;

The hfwDiskStat function stores the drive status in the area specified by phfwDiskStatus. The following explains the parameters of this function.

## phfwDiskStatus:

Specifies a pointer to the following HFW\_DISK\_STATUS struct for storing the drive status.

```
typedef struct HFW_DISK_STATUS{
    DWORD   Disk_Count;
    DWORD   Disk_Status[16]; // Drive status
} HFW_DISK_STATUS, *PHFW_DISK_STATUS;
```

Disk\_Count contains the number of drive bays (2) of this device.

Disk\_Status[n] contains the status of the drive in drive bay (n+1). The following shows values indicating each status. The most significant 16 bits are for future use. Since the values are uncertain, do not access them.

Table 6-9 Values for Disk\_Status

define value	Meaning
DISKSTAT_HEALTHY (0x00000001)	The drive is running normally.
DISKSTAT_SMART (0x00000008)	Drive failure prediction (SMART) was detected. (*1)
DISKSTAT_NOT_CONNECTED (0x00000010)	The drive is not connected.
DISKSTAT_UNKNOWN (0x00000020)	The drive status cannot be obtained.
DISKSTAT_OVERRUN (0x00000040)	The drive usage time has exceeded the predetermined value. (*2)
DISKSTAT_OFFLINE (0x00000002)	The drive is disconnected from RAID. (*3)
DISKSTAT_REBUILD (0x00000004)	The drive newly connected to RAID is under rebuild. (*3)

(\*1) No value is returned if drive failure prediction feature is not enabled.

(\*2) No value is returned if drive usage time monitoring feature is not enabled.

(\*3) The value is returned only for the RAID model.



## 6. LIBRARY FUNCTIONS

If multiple statuses occur at the same time, the logical sum of the values shown in Table 6-9 is returned.

- If drive failure prediction detection and excess of usage time occur at the same time: DISKSTAT\_SMART | DISKSTAT\_OVERRUN (0x48) is returned.

### <Diagnostics>

If this function terminates normally when called, TRUE is returned. If it terminates with an error, FALSE is returned. In the case of error termination, the value stored in phfwDiskStatus is invalid. Also, in the case of error termination, the Windows API GetLastError function can be called to obtain the error code. The error codes returned by this function are as follows.

Error code (value)	Description
HFWRAS_INVALID_PARAMETER (0x20000001)	The specified argument is incorrect.
HFWRAS_NOT_INITIALIZE (0x20000002)	RAS software has not finished starting up.
HFWRAS_INTERNAL_ERROR (0x20000003)	An internal error has occurred.

Any error codes other than the above are from the Windows API function used by this function. For details, see the help for the Windows API.

### <Sample program>

A sample program in C is provided as a usage example of this function. For details about the storage location and file name of this sample program, see “6.1.12 Sample programs.”

## 6.1.10 Functions for obtaining RAID status

## (1) hfwRaidStat

&lt;Name&gt;

hfwRaidStat - Obtaining RAID status

&lt;Format&gt;

#include &lt;hfwras.h&gt;

BOOL hfwRaidStat (PHFW\_RAID\_STATUS phfwRaidStatus);

&lt;Description&gt;

The hfwRaidStat function stores the RAID status in the area specified by phfwRaidStatus.

The following explains the parameters of this function.

phfwRaidStatus:

Stores the RAID status. This specifies a pointer to the HFW\_RAID\_STATUS struct shown below.

```
typedef struct HFW_RAID_STATUS{
    DWORD          Array_Count;      // Number of arrays to be monitored
    HFWARRAY_STATUS Array[16];      // Array information
} HFW_RAID_STATUS, *PHFW_RAID_STATUS;
```

Array\_Count contains the number of RAID arrays to be monitored in this device.

In each array element, a RAID status value is set. In this case, the number of valid elements is the value returned by Array\_Count (values up to Array[0] are valid). Any subsequent elements are for future use. Since values of such elements are NULL, do not access them. The HFWARRAY\_STATUS structure is as shown below.

```
typedef struct HFW_ARRAY_STATUS{
    DWORD          Level;            //RAID level
    DWORD          Disk_Number;      //RAID configuration drives
    DWORD          Status;           //RAID status
    DWORD          Progress;         //Progress state (0 to 99[%])
}HFW_ARRAY_STATUS, *PHFW_ARRAY_STATUS ;
```

## 6. LIBRARY FUNCTIONS

“Level” contains the RAID level. Its value is as shown below.

Table 6-10 Value for Level of the HFW\_ARRAY\_STATUS Struct

No.	define value	Meaning
1	HFW_RAID1 (value is 0x00000001)	RAID1 is configured.

In “DiskNumber”, the value indicating the drive bay of RAID is stored. The value is a logical sum of the following values.

Table 6-11 Value for DiskNumber of the HFW\_ARRAY\_STATUS Struct

No.	define value	Meaning
1	HFW_DISK1 (value is 0x00000001)	Drive bay 1 is contained in RAID configuration.
2	HFW_DISK2 (value is 0x00000002)	Drive bay 2 is contained in RAID configuration.

“Status” contains the value indicating the RAID status. The value is a logical sum of the following values. No. 1 to No. 4 indicate RAID status, and No. 5 and No. 6 show the detailed information.

Table 6-12 Value for Status of the HFW\_ARRAY\_STATUS Struct

No.	define value	Meaning
1	HFW_RAID_OPTIMAL (value is 0x00000001)	Normal operation.
2	HFW_RAID_DEGRADE (value is 0x00000004)	Degraded operation.
3	HFW_RAID_UNKNOWN (value is 0x00000008)	Status is unknown.
4	HFW_RAID_FAIL (value is 0x00000010)	Failure.
5	HFW_RAID_REBUILD (value is 0x00010000)	There is a drive being rebuilt.
6	HFW_RAID_MEDIA_ERROR (value is 0x00020000)	Media error occurred.

When a drive is under rebuild, for example, the RAID status is in degraded operation (HFW\_RAID\_DEGRADE) and the detailed information indicates that there is a drive being rebuilt (HFW\_RAID\_REBUILD). Therefore their logical sum “HFW\_RAID\_DEGRADE|HFW\_RAID\_REBUILD” is stored in “Status”.

Possible combinations of RAID status and detailed information are as shown below.

Table 6-13 Combination of Values for Status of the HFW\_ARRAY\_STATUS Struct

No.	RAID status	Detailed info
1	HFW_RAID_OPTIMAL	None
2		HFW_RAID_MEDIA_ERROR
3	HFW_RAID_DEGRADE	None
4		HFW_RAID_MEDIA_ERROR
5		HFW_RAID_REBUILD
6		HFW_RAID_REBUILD   HFW_RAID_MEDIA_ERROR
7	HFW_RAID_UNKNOWN	None
8	HFW_RAID_FAIL	None

No. 8: Since the system cannot operate normally, the value may not be obtained.

“Progress” contains the progress state of rebuilding. If the RAID status value does not contain HFW\_RAID\_REBUILD, “0” is stored.

#### <Diagnosis>

If this function is called and terminates normally, TRUE is returned. If it terminates with an error, FALSE is returned. In the case of error termination, the value stored in phfwRaidStatus is invalid. Also in the case of error termination, the GetLastError function of Windows API can be called to obtain the error code. The error codes returned by this function are as follows.

Error code (value)	Description
HFWRAS_INVALID_PARAMETER (0x20000001)	The specified argument is incorrect.
HFWRAS_NOT_INITIALIZE (0x20000002)	RAS software has not finished starting up.
HFWRAS_INTERNAL_ERROR (0x20000003)	An internal error has occurred.
HFWRAS_NOT_RAID (0x20000101)	The device executing this function is not an RAID model.

Any other error codes than the above are from Windows API functions used by this function. For details, see the help of Windows API.

#### <Sample program>

A sample program in C is provided as a usage example of this function. For details about the storage location and file name of this sample program, see “6.1.12 Sample programs.”

## 6. LIBRARY FUNCTIONS

### (2) RaidStat

<Name>

RaidStat - Obtaining the status of RAID configuration drives

<Format>

```
#include <hfwraid.h>
```

```
BOOL RaidStat(PRAID_DATA pRaidData);
```

<Description>

The RaidStat function stores the status of RAID configuration drives in the area specified by pRaidData. The following explains the parameters of this function.

pRaidData:

This specifies a pointer to the RAID\_DATA struct shown below to store the obtained drive status.

---

#### NOTE

---

Windows® 7 (64-bit) does not support this function. Use the function provided as an alternative (hfwRaidStat).

---

```
typedef struct RAID_DATA {  
    DWORD Hdd1_Status; //Drive status of drive bay 1  
    DWORD Hdd2_Status; //Drive status of drive bay 2  
} RAID_DATA, *PRAID_DATA;
```

In the elements of the RAID\_DATA struct, the values shown in Table 6-14 are stored when this function terminates successfully.

Table 6-14 Values for Elements of the RAID\_DATA Struct

No.	Value	Meaning
1	HDD_ONLINE	Normal operation.
2	HDD_OFFLINE	The drive is disconnected from RAID.
3	HDD_REBUILD	RAID is under rebuild (being copied).
4	HDD_REMOVED	Not connected.
5	HDD_UNKNOWN	Status is unknown.

## &lt;Diagnosis&gt;

If this function is called and terminates normally, TRUE is returned. If it terminates with an error, FALSE is returned. In the case of error termination, the value stored in pRaidData is invalid. Also in the case of error termination, the GetLastError function of Windows API can be called to obtain the error code. The error codes returned by this function are as follows.

Error code (value)	Description
RAID_INVALID_PARAMETER (0x2001)	The specified argument is incorrect.
RAID_GET_STATUS_ERROR (0x2002)	An internal error has occurred.
NOT_RAID (0x2006)	The device executing this function is not an RAID model.
RAID_NOT_READY (0x3001)	RAS software has not finished starting up.

Any other error codes than the above are from Windows API functions used by this function. For details, see the help of Windows API.

## 6. LIBRARY FUNCTIONS

### 6.1.11 Functions for controlling the digital LED for status indication

#### (1) Application status code display function (SetStCode7seg)

<Name>

SetStCode7seg - Displaying the application status

<Format>

```
#include <ctrl7seg.h>
```

```
BOOL SetStCode7seg(DWORD dwStCode);
```

<Description>

This function displays the application status code on the digital LED for status indication. The digital LED for status indication displays the value specified by this function in hexadecimal.

The following explains the parameters of this function.

dwStCode:

Specifies the application status code to be displayed on the LED, from 0 to 255. If any other values are set, error termination occurs, and the CTRL7SEG\_INVALID\_PARAMETER error code is returned when the Windows API GetLastError function is called.

<Diagnostics>

If this function terminates normally when called, TRUE is returned. If it terminates with an error, FALSE is returned.

In the case of error termination, the Windows API GetLastError function can be called to obtain the error code. The error codes returned by this function are as follows.

Error code (value)	Description
CTRL7SEG_INVALID_PARAMETER (0x2001)	The specified argument is incorrect.

Any error codes other than the above are from the Windows API function used by this function. For details, see the help for the Windows API.

<Sample program>

A sample program in C is provided as a usage example of this function. For details about the storage location and file name of this sample program, see “6.1.12 Sample programs.”

## (2) Function for hiding the application status code (TurnOff7seg)

## &lt;Name&gt;

TurnOff7seg - Hiding the application status code

## &lt;Format&gt;

```
#include <ctrl7seg.h>  
BOOL TurnOff7seg(VOID);
```

## &lt;Description&gt;

This function hides the application status code displayed on the digital LED for status indication. When this function is called, the digital LED for status indication turns off.

## &lt;Diagnostics&gt;

If this function terminates normally when called, TRUE is returned. If it terminates with an error, FALSE is returned.

In the case of error termination, the Windows API GetLastError function can be called to obtain the error code. The error code is that of the Windows API function used by this function. For details, see the help for the Windows API.

## &lt;Sample program&gt;

A sample program in C is provided as a usage example of this function. For details about the storage location and file name of this sample program, see “6.1.12 Sample programs.”



## 6. LIBRARY FUNCTIONS

### (3) Status display mode settings function (SetMode7seg)

<Name>

SetMode7seg - Setting the status display mode

<Format>

```
#include <ctrl7seg.h>
BOOL SetMode7seg(DWORD dwMode);
```

<Description>

This function sets the status display mode of the digital LED for status indication. The following explains the parameters of this function.

dwMode:

Specifies the “status display mode” of the digital LED for status indication. Table 6-15 shows the values specified of dwMode.

Table 6-15 Values Specified for dwMode of the SetMode7seg Function

dwMode	Description
RASST_MODE	Sets the hardware status display mode.
APPST_MODE	Sets the application status display mode.

If any value other than the above is specified, this function terminates with an error, and the CTRL7SEG\_INVALID\_PARAMETER error code is returned when the Windows API GetLastError function is called.

<Diagnostics>

If this function terminates normally when called, TRUE is returned. If it terminates with an error, FALSE is returned.

In the case of error termination, the Windows API GetLastError function can be called to obtain the error code. The error codes returned by this function are as follows.

Error code (value)	Description
CTRL7SEG_INVALID_PARAMETER (0x2001)	The specified argument is incorrect.

Any error codes other than the above are from the Windows API function used by this function. For details, see the help for the Windows API.

<Sample program>

A sample program in C is provided as a usage example of this function. For details about the storage location and file name of this sample program, see “6.1.12 Sample programs.”

## 6.1.12 Sample programs

The sample C programs for using RAS library functions are stored in the %ProgramFiles%\HFWRAS\sample directory. Refer to them when developing programs and checking operation.

The following lists the sample programs.

Table 6-16 List of Provided Sample Programs

No.	File name	Contents
1	shutd.c	Sample program for the BSSysshut function
2	wdt.c	Sample program for the WdtControl function
3	gendoex.c	Sample program for the GendoControlEx function
4	gendiex.c	Sample program for the GetGendiEx function
5	Mcon.c	Sample program for the MconWriteMessage function
6	MemErr.c	Sample program for the GetMemStatus function
7	hfwdiskstat.c	Sample program for the hfwDiskStat function
8	7seg.c	Sample program for the digital LED for status indication control functions (SetStCode7seg function, TrunOff7seg function, and SetMode7seg function)
9	Fanerr.c	Sample program for RAS event acquisition (for details about RAS event notification feature, see “4.2 RAS Event Notification Feature [Model 35 Only]”)
10	hfwRaidStat.c	Sample program for the hfwRaidStat function

**This Page Intentionally Left Blank**

## CHAPTER 7 MAINTENANCE AND FAILURE ANALYSIS

### 7.1 STOP Error Code Cause Notification Feature

#### 7.1.1 Overview

This device saves the contents of memory in a memory dump file when forced recovery from an OS freeze or hardware-related NMI is performed, or when an uncorrectable memory error occurs. In this case, a blue screen occurs, and the 0x00000080 STOP error code (herein as 0x80) is displayed.

The 0x80 STOP error code can indicate multiple STOP error causes, but in some cases, the information displayed below the STOP error code can be used to identify the cause, or make the area to be investigated more clear.

This feature detects blue screens due to the 0x80 STOP error code, and logs the cause of the blue screen to the event log.

#### 7.1.2 Target STOP error causes

This feature runs when a blue screen occurs due to one of the causes shown in Table 7-1. Other STOP errors of Windows® are not supported.

Table 7-1 List of Target STOP Error Causes

Cause	STOP error code	Contents displayed for the STOP error screen
NMI occurrence due to the reset button or reset signal input to the RMTRESET external contact	0x00000080	Hardware malfunction.  == Detailed Information == 0x9201: IOCHK Error.
NMI occurrence due to a parity error in the PCI, LPC, or other bus	0x00000080	Hardware malfunction.  == Detailed Information == 0x9202: PCI Bus Parity Error.
Uncorrectable memory error	0x00000080	Hardware malfunction.  == Detailed Information == 0x????: XXXXXXXXXXXX (*)

(\*) The following contents are displayed depending on the cause.

0x9217: Uncorrectable Error at DIMM B.

0x9218: Uncorrectable Error at DIMM A.

For details about how to handle blue screens, refer to CHAPTER 6 in the *HF-W2000 Model 35/30 INSTRUCTION MANUAL* (manual number WIN-62-0060).

## 7. MAINTENANCE AND FAILURE ANALYSIS

### 7.1.3 Event logs

Table 7-2 shows the event logs recorded by this feature. These event logs are recorded in the system logs.

Table 7-2 Recorded Event Logs

Event ID	Source	Type	Category	Description
800	HFWRAS_SYS	Information	HFWRAS	%1 Detailed code = %2.

(Note 1) The following messages are saved in %1 above, depending on the detailed information.

1. Reset signal was input.
2. PCI Bus Parity Error.
3. Uncorrectable Error at DIMM A.
4. Uncorrectable Error at DIMM B.

(Note 2) One of the following hexadecimal codes is saved in %2, according to %1.

1. 0x9201
2. 0x9202
3. 0x9217
4. 0x9218

## 7.2 Log Information Collection Window **[Model 35 Only]**

### 7.2.1 Overview

The log information collection window allows the following operations to be performed in a GUI environment.

(1) Collection of log information data

Data is saved for use in preventative maintenance and post-hoc analysis of troubles. The data is compressed and saved as a single file (logsave.dat).

(2) Collection of memory dump files

Memory dump files are collected by the OS, for which data is saved as a compressed file (memory.mcf). At this time, a small memory dump file called a “minidump file” is also collected.

---

#### NOTE

- This feature is not supported for Model 30.
  - CPU load increases while memory dump files are collected. Since this may interfere with user application operation, do not use the log information collection window to collect memory dump files while this device is running.
- 

### 7.2.2 Opening the log information collection window

Perform the following to open the log information collection window.

Administrator privilege is needed to use this window. Log on using a computer administrator account before opening the window.

1. Click the [Start] button.
2. Choose [All Programs], and then [RAS Software].
3. Click [RAS Maintenance Support].
4. If the [User Account Control] window appears, click the [Yes] button.  
(For Windows® 7 only)

---

#### NOTE

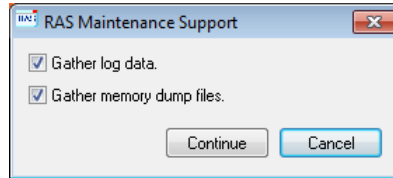
Since the log information collection window cannot be used by multiple users at once, an error will occur if the log information collection window is used from multiple consoles, such as through user switching. As such, close any log information collection windows open in other consoles before opening the window.

---

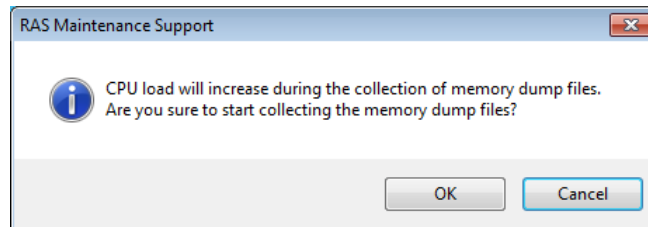
## 7. MAINTENANCE AND FAILURE ANALYSIS

### 7.2.3 Using the log information collection window

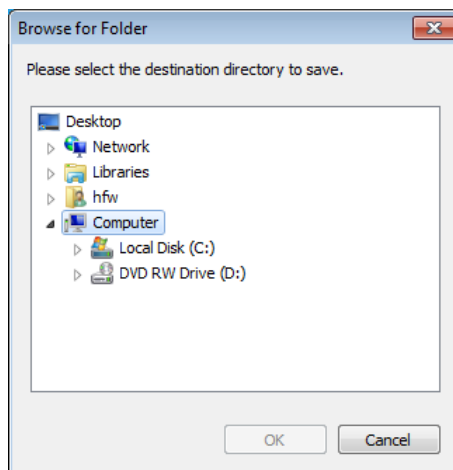
1. Display the log information collection window. Since the [Gather log data] and [Gather memory dump files] checkboxes are both selected by default, clear checkboxes for any information that does not require collection, and click the [Continue] button.



2. If [Gather memory dump files] is selected, the following dialog box is displayed, so click the [OK] button. Alternately, click the [Cancel] button to return to the log information collection window without performing maintenance.



3. The following dialog box is displayed. Specify the save destination directory, and click the [OK] button. To cancel operation, click the [Cancel] button to return to the log information collection window without performing maintenance.







**7.3 Trend Log of Internal Cubicle Temperatures [Model 35 Only]**

7.3.1 Overview

This feature regularly obtains the internal cubicle temperature of this device, and records it in a log file. The log cycle settings command can be used to tune the log cycle for internal cubicle temperature information. The initial setting of log cycle is 60 minutes, and can be set to one of 10 minutes, 30 minutes, and 60 minutes.

---

**NOTE**

---

This feature is not supported for Model 30.

---

7.3.2 Log files

Internal cubicle temperature information is recorded to log files at the interval of the specified log cycle. If this device runs continuously for 8 or more hours, the highest temperature and lowest temperature of every 8 hours is also written to the log file. When the log becomes full in each file, the first case is overwritten.

Table 7-3 lists the names of the log files recorded.

Table 7-3 Recorded Log Files

Storage folder	File name	Description
%ProgramFiles%\HFWRAS\log	temp.csv	The internal cubicle temperature is logged for each logging cycle. (Up to 51200 cases)
	temp_mm.csv	The highest temperature and lowest temperature of every 8 hours are logged. (Up to 1100 cases)

<Viewing log information>

The content of a log file can be viewed by opening the file with Notepad or similar application. Since the csv format is used, log information can also be read by spreadsheet software and database software, and displayed in graphs.

Note that these log files can be collected by using the logsave command. For details about using the logsave command, refer to CHAPTER 7 in the *HF-W2000 Model 35/30 INSTRUCTION MANUAL* (manual number WIN-62-0060).

<Log information format>

The log information formats are as follows.

(1) temp.csv

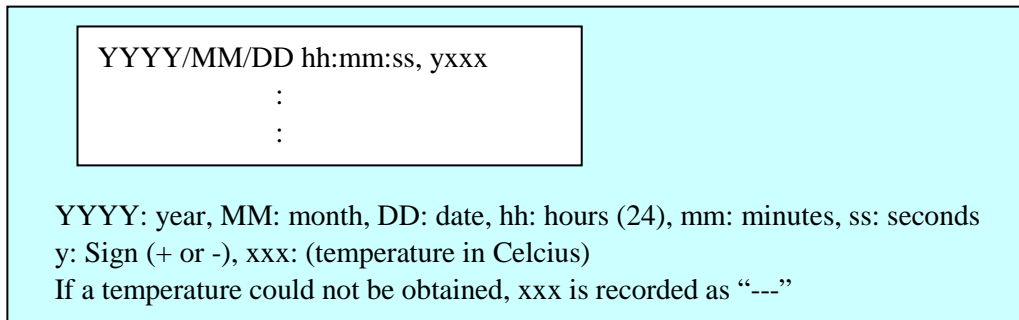


Figure 7-2 Log Information Format 1

(2) temp\_mm.csv

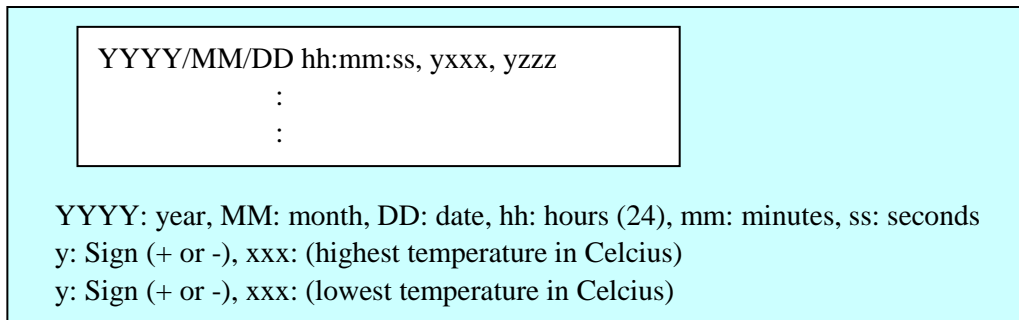


Figure 7-3 Log Information Format 2

## 7. MAINTENANCE AND FAILURE ANALYSIS

### 7.3.3 Command for setting the temperature log cycle

<Name>

tmplogset - Setting the logging cycle

<Format>

tmplogset

<Description>

This command sets the log cycle for trend log feature for logging internal cubicle temperatures. The following explains how to use this command.

#### 1. Open a command prompt.

Administrator privilege is needed to execute this command. Log on using a computer administrator account before opening the command prompt.

- For Windows® XP
  - Click the [Start] button.
  - Choose [All Programs], and then [Accessories].
  - Click [Command Prompt].

- For Windows® 7

If the user account control (UAC) is enabled, follow the procedures in “APPENDIX A RUNNING THE COMMAND PROMPT AS AN ADMINISTRATOR” to use the command prompt as an administrator.

#### 2. Execute the tmplogset command in the command prompt to display the following initial screen with the current settings.

```
>tmplogset
Logging time of the cycle: 60 minutes
1. Change at logging cycle [10, 30, 60 minutes]
2. Exit
:_
```

3. Enter “2” in the initial screen to terminate the tmplogset command without changing any settings. Otherwise, enter “1” and press the [Enter] key to output the following message.

```
Please select new time of the cycle.  
When the return is input, it becomes like a present setting.  
1. 10 minutes  
2. 30 minutes  
3. 60 minutes  
:_
```

4. Enter the number for the cycle time to be set, and press the [Enter] key. If the entered value is out of range, the following message is displayed, prompting re-input of a correct value.

```
The entered setting is invalid.  
Please enter a setting again. [input range: 1-3]
```

5. Enter a value from 1 to 3 to display the following message. Here, “x” is the value entered.

```
New logging time of the cycle is x.  
Is this value set?(y-YES/n-NO)  
:_
```

6. Enter “y” and press the [Enter] key to set up a new log cycle time and terminate the command. These take effect when the command terminates. To confirm the changed settings, execute this command again and see the initial screen.  
To change the log cycle, enter “n” and press the [Enter] key. The following message is displayed, and the command terminates without changing any settings.

```
The setting takes no effect, because you enter the letter ‘n’.
```

## 7. MAINTENANCE AND FAILURE ANALYSIS

Note that if no administrator privilege exists during command execution, the following message is displayed and the command terminates.

- For Windows® XP

```
>tmplogset
You do not have the privilege to perform this command.
Please log on as a group of administrators and perform this command again.
```

- For Windows® 7

```
>tmplogset
You do not have the privilege to run this command.
Please run this command again on "Administrator: Command Prompt".
```

If an internal error occurs during command execution, the following message is displayed and the command terminates.

```
Error: Systemcall failed. (API Name : Error Code)
```

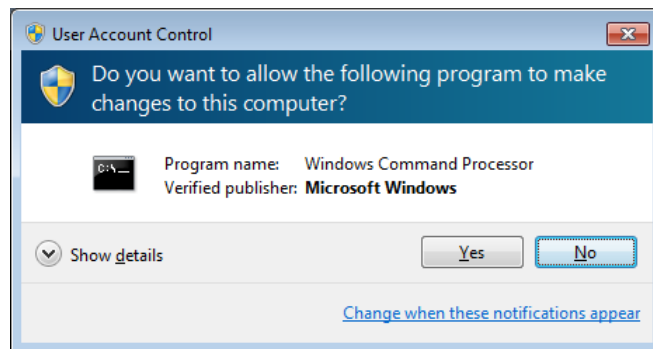
In the above message, the Windows API in which the error occurred is displayed for "API Name", and a hexadecimal number indicating the error code is displayed for "Error Code". If the above message is displayed, execute the command again.

## APPENDIX A RUNNING THE COMMAND PROMPT AS AN ADMINISTRATOR

Perform the following to run the command prompt as an administrator.

1. Log on using a computer administrator account.
2. Click the [Start] button.
3. Choose [All Programs].
4. Choose [Accessories].
5. Right click [Command Prompt], and then choose [Run as administrator].  
(If the administrator account used for logon is the built-in Administrator account, click [Command prompt].)

When the [User Account Control] window appears, click the [Yes] button.



**This Page Intentionally Left Blank**