WIN－63－0092

HITACHI

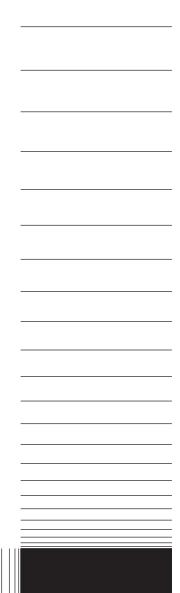## HITACHI INDUSTRIAL COMPUTER

HF-W2000 Model 48/45

RAS FEATURES MANUAL

USER'S
MANUAL

HITACHI

HITACHI INDUSTRIAL COMPUTER

HF-W2000 Model 48/45

RAS FEATURES MANUAL

Read and keep this manual.
• Read safety instructions carefully and understand them before starting your operation. • Keep this manual at hand for reference.

USER'S
MANUAL

TP<IC> (FL-MW2007)

⚠ SAFETY INSTRUCTIONS

Carefully read and fully understand the safety precautions below before operating the equipment.

● Operate the equipment by following the instructions and procedures described in this manual.

● Pay attention especially to safety precautions displayed on the equipment or in this manual. Make sure you follow them. Otherwise, personal injury and property damage including damage to the equipment may result.

● A safety precaution is indicated by a heading as shown below. A heading is either a safety alert symbol; a word such as "DANGER", "WARNING", "CAUTION", or "NOTICE"; or a combination of both.

⚠ This is a safety alert symbol. This symbol is used to signify potential hazards that may result in personal injury or death. Make sure you follow the safety message that follows this symbol in order to avoid possible injury or death.

⚠ DANGER: This symbol is used to indicate imminent hazards that will highly likely result in serious personal injury or death.

⚠ WARNING: This symbol is used to indicate potential hazards that may result in serious personal injury or death.

⚠ CAUTION: This symbol is used to indicate potential hazards that may result in minor or moderate personal injury.

NOTICE: This symbol is used to indicate hazards that may result in equipment or property damage but not personal injury.

The heading "NOTE" is used to indicate a cautionary note about handling and operation of the equipment.

● Do not attempt to perform any operation that is not described in this manual. If there is any problem with the equipment, call your maintenance personnel.

● Read this manual carefully and fully understand the directions and precautions written in this manual before operating the equipment.

● Keep this manual nearby so that you can reference the manual anytime you need it.

● Every effort has been made to specify the best precautions on the equipment and in the manual. Nevertheless, unexpected incidents may occur. When you use the equipment, you are asked not only to follow the instructions but also to use your own judgement on safety.

⚠ SAFETY INSTRUCTIONS (Continued)

1. COMMON SAFETY PRECAUTIONS
   Carefully read and fully understand the following safety precautions.

   1.1 ⚠ WARNING

   ● This equipment is not designed and manufactured to be used for a life-critical system that requires extreme safety. If there is a possibility that the equipment may be used for this purpose, contact one of our sales representatives.

   ● In case of smoke, a burning smell, or the like, turn off the power to the equipment, disconnect the power cable from the outlet, and contact your sales representative or maintenance personnel. Using the faulty equipment without repair may result in a fire or an electric shock.

   ● This equipment has built-in magnetic disk drives. Do not hit the equipment or give a shock or vibration to the equipment because that may cause the equipment to fail. Should you drop the equipment or damage its chassis, disconnect the power cable from the outlet and contact your maintenance personnel. Using the faulty equipment without repair may result in a fire or an electric shock. Do not give a shock to the equipment when unpacking or carrying the equipment.

   ● Do not modify this equipment because that may result in a fire or an electric shock. In no event will Hitachi be responsible for any consequences that results from the modification or alteration of the equipment.

⚠ SAFETY INSTRUCTIONS (Continued)

1.2 ⚠ CAUTION

● If the equipment drops or is tipped over, personal injury may result. Pay full attention when transporting the equipment.

● Make sure you do not catch or hit your fingers to cause personal injury when unpacking or carrying the equipment.

1.3 *NOTICE*

● This equipment alone cannot guarantee the system safety. In order to ensure sufficient safety of your system even when this equipment should fail, malfunction, or have program bugs, you must add systemic protections such as building external protective/safety circuits to facilitate safety measures to prevent personal injury and serious accidents.

● Use the basic software that we specify. Operation is not guaranteed if any other basic software is used.

● Do not use the equipment in the environment full of dust or with corrosive gas because that may cause the equipment to fail.

● Make sure sufficient clearance is provided for air intake and exhaust in front of and behind the equipment. Otherwise, the temperature inside the equipment may rise and that may cause a failure or short life span of the equipment. In addition, you need to ensure sufficient clearance for maintenance work.

● Performing emergency shutdown (that is, unplugging the power cable from the outlet or shutting off the circuit breaker without proper shutdown of the OS) may cause the OS or applications not to work properly or may cause the saved data to be corrupted. Never perform emergency shutdown unless you must stop the system immediately due to some kind of error.

● Keep in mind that if the power supply is cut, the system may not be able to recover automatically.

2. SAFETY WARNINGS IN THIS MANUAL

  2.1 Safety Warning Indicated as "*NOTICE*"

  ● When failure of a drive is anticipated, the drive may experience hardware
    failure in near future. We recommend you to back up the data and replace the
    drive. For information about how to replace a drive, refer to "HF-W2000
    Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".
    (See page 2-3.)

  ● When OS hung-up occurs, processes on the OS cannot run as scheduled, and
    the facility using this equipment may be affected due to delay in processing.
    If OS hung-up occurs, resolve it immediately.
    (See page 2-6.)

  ● If this equipment continues to operate while fan failure is detected, internal
    parts such as a processor are not cooled down sufficiently, and that may
    cause thermal runaway of the system due to malfunction of the equipment or
    result in damage to the parts.
    (See page 3-4.)

  ● When the **Hardware status** window shows an error in hardware, resolve the
    error immediately.
    (See page 4-2.)

  ● When failure of a drive is anticipated, the drive may experience hardware
    failure in near future. We recommend you to back up the data and replace the
    drive. For information about how to replace a drive, refer to "HF-W2000
    Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".
    (See page 4-9.)

  ● If this equipment continues to operate with fan failure detected, internal parts
    such as a processor are not cooled down sufficiently, and that may cause
    thermal runaway of the system due to malfunction of the equipment or result
    in damage to the parts. If possible, enable the automatic shutdown feature.
  ● If the automatic shutdown feature is not used, have a user application detect a
    fan failure using a RAS event and shut down the equipment.
    (See page 5-2.)

● Log function exits (asynchronously) without waiting for the data to be actually written to a log file. That means this function does not return an error even when writing to a log file fails for some reason. We recommend important information be recorded in the OS event log.

(See page 6-15.)

● CPU load increases while memory dump files are being collected. While CPU load is high, operation of user applications can be disturbed. Make sure that you do not collect memory dump files using the log information collection window while the applications for business use are running on this equipment.

(See page 7-4.)

● While the equipment is running in simulation mode, monitoring of the actual hardware status is disabled. Errors including fan failure and abnormal temperature cannot be detected. You must never use this equipment in simulation mode for business use. Use the simulation function only for testing a user application and checking the notification interface of the RAS software.

(See page 8-2.)

PREFACE

This manual describes how to use the Reliability, Availability, Serviceability (RAS) feature of the HITACHI INDUSTRIAL COMPUTER HF-W2000 Model 48/45 (hereinafter referred to as "this equipment").

<Organization of this manual>
  This manual is organized as follows.
  CHAPTER 1   CAPABILITIES OF THE RAS FEATURE
  CHAPTER 2   ITEMS MONITORED BY THE RAS FEATURE
  CHAPTER 3   SETTING UP THE RAS FEATURE
  CHAPTER 4   CHECKING THE HARDWARE STATUS
  CHAPTER 5   CONTROLLING THE HARDWARE
  CHAPTER 6   LIBRARY FUNCTIONS
  CHAPTER 7   FEATURES RELATED TO MAINTENANCE AND FAILURE ANALYSIS
  CHAPTER 8   SIMULATING THE HARDWARE STATUS

<Precautions when you use the RAS feature>
  ● Precautions about a RAS external contacts interface
    This manual provides an explanation about a RAS external contacts interface. Note that this interface is an optional feature.

  ● Precautions about an event log entry at the startup of the SNMP service
    When you enable the SNMP service, a standard feature of Windows®, in order to use remote notification, an error log entry with event ID 1500 may be recorded at the startup of the SNMP service. This event log entry is recorded when SNMP trap notification has not been set up. Set up the trap notification according to "4.5.3   Enabling the remote notification".

  ● User Account Control
    If User Account Control (UAC) is enabled in Windows® configuration, User Account Control windows is displayed when you run application or command. In this case, click **OK** or **Continue**.

<Trademarks>
  • Microsoft®, Windows®, Windows Server®, Windows NT®, and Visual Basic® are trademarks or registered trademarks of U.S. Microsoft Corporation in the United States and other countries.
  • All other product names (software and hardware) not from Hitachi described in this manual are registered trademarks, trademarks, or products of their respective owners.

<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

　　1 TB (terabyte) = 1,099,511,627,776 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

　　1 TB (terabyte) = $1,000^4$ bytes

<Note for replacing terms for Windows® 10>

This manual describes user actions in Windows®. The terms "log on" and "log off" in Windows® 7 and Windows Server® 2008 R2 have been renamed "sign in" and "sign out" respectively in Windows® 10. If the OS you use is Windows® 10, replace the former terms in this manual with the latter terms as necessary.

In addition, this manual describes remote notification feature in Windows®. The terms "Allowed a program through Windows Firewall" and "Allowed programs" in Windows® 7 and Windows Server® 2008 R2 have been renamed "Allowed an app through Windows Firewall" and "Allowed apps" in Windows® 10. If the OS you use is Windows® 10, replace the former terms in this manual with the latter terms as necessary.

<Note for display images>

The display images of this manual assume Windows® 7 but it is similar under other OS.

CONTENTS

FIGURES

TABLES

CHAPTER 1   CAPABILITIES OF THE RAS FEATURE

The HF-W series come with the Reliability, Availability, Serviceability (RAS) feature that you would expect from a highly reliable industrial computer.
The following is an overview of the RAS feature.

Table 1-1   Overview of the RAS Feature

Category		Item
Monitoring		Hardware status monitoring
		OS hung-up monitoring
		Watchdog timer monitoring
GUI function setting		RAS setup window
Status check	GUI output	Hardware status window
	Notification	Event notification
		Popup notification
		Status display digital LEDs
		Remote notification
		Status acquisition using library functions
Control	Shutdown /Startup suppression	Automatic shutdown
		Shutdown using library functions
		Startup suppression when severe failure occurs
		Controlling the general purpose external contacts
		Controlling the status display digital LEDs
Library functions		RAS library
Maintenance/ Failure analysis	Memory dump related	Memory dump collection
		Error cause notification with STOP error code
		Log information collection window
		Maintenance operation support commands
		Logging the trend of the temperature inside the chassis
Simulation		Hardware status simulation

<Monitoring>
(1) Hardware status monitoring

Monitors the hardware status of this equipment including the status of the fans and drives as well as the temperature inside the chassis.

(2) OS hung-up monitoring

Monitors the operational state of the OS using a dedicated timer implemented on this equipment. As long as a process with the highest priority (real-time priority class) can run properly, the status lamp on the front of this equipment is lit green.

(3) Watchdog timer monitoring

Monitors whether processes are scheduled properly, using the watchdog timer implemented on this equipment. This function also offers library functions to use the watchdog timer.

<GUI function setting>
(4) RAS Setup window

Provides a graphical user interface for configuring RAS Feature settings including the condition of automatic shutdown and the setting of the watchdog timer.
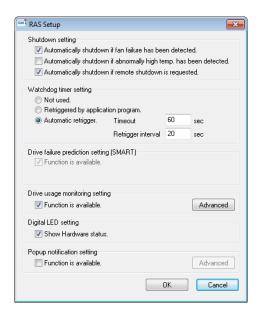
Figure 1-1 RAS Setup Window

<Status check>

  (5) Hardware status window

     Displays the hardware status of this equipment using a graphical interface. There is always
     an icon in the notification area of the taskbar to display the hardware status.

Figure 1-2   Hardware Status Icon

  (6) Event notification

     Enables a user application to check the hardware status of this equipment by monitoring the
     status of event objects.

  (7) Popup notification

     Notifies a user that an error occurred in the hardware of this equipment by displaying popup
     messages.

  (8) Status display digital LEDs

     The status display digital LEDs are on the front of this equipment and notify a user that an
     error occurred in the hardware of this equipment. These LEDs can be used by a user
     application in order to, for example, notify the failure of the application.

  (9) Remote notification

     A remote device can check the hardware status of this equipment. A remote device can also
     be notified whenever the hardware status changes.

  (10) Status acquisition using library functions

     This function enables a user application to get the hardware status of this equipment by
     using the RAS library.

<Control>

  (11) Automatic shutdown

     Automatically shuts down the equipment when fan failure, abnormal temperature inside
     the chassis, or a remote shutdown signal input is detected. Use "(4) RAS Setup window" to
     enable or disable the automatic shutdown function.

(12) Shutdown using library functions

This function enables a user application to shutdown this equipment by using the RAS library.

(13) Startup suppression when severe failure occurs

Suppresses startup of this equipment when failure such as fan failure is detected during OS startup in order to protect the hardware.

(14) Controlling the general purpose external contacts and the status display digital LEDs

Enables a user to control the general purpose external contacts and the status display digital LEDs with the RAS library.

Four input and three output general purpose external contacts are available for a user. If you use those contacts, signals can be input from an external device to this equipment, and signals can be output from this equipment to an external device.

<Library functions>

(15) RAS library interface

Offers library functions for recording log information in addition to the library functions offered by (10), (12), and (14).

<Maintenance / Failure analysis>

(16) Memory dump collection

This function records the contents of the system memory in a file (memory dump file) when the reset switch is pressed after failure occurred, for example, after the equipment stops unexpectedly. By analyzing the data in this memory dump, you can investigate the cause of the failure.

(17) Error cause notification with STOP error code

This function detects a blue screen caused by an error with the STOP error code 0x80 and records the cause of the blue screen in the event log.

(18) Log information collection window

Allows you to collect log data and memory dump files for this equipment using a graphical user interface.

(19) Maintenance operation support commands

These commands include a command used for saving failure information such as memory dump files and event log files to an external medium.

(20) Logging the trend of the temperature inside the chassis

This function periodically measures the temperature inside the chassis of this equipment and records the data into a file.

<Simulation>

(21) Hardware status simulation

Simulates the hardware status of this equipment. By using this function, you can test a user application and check the notification interface of the RAS software without actual hardware failure.

This manual explains functions (1) through (12), (14), (15), (17), (18), (20), and (21). For details about other functions, refer to "HF-W2000 Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".

This page is intentionally left blank.

CHAPTER 2　ITEMS MONITORED BY THE RAS FEATURE

This chapter explains the items monitored by the RAS Feature.

For information about the hardware specifications of the RAS external contacts interface (optional) described in this chapter and the usage of each contact, refer to "HF-W2000 Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".

2.1　Fan Monitoring

The fan monitoring function monitors the multiple fans located in this equipment and notifies in the following methods when one of the fans malfunctions.

(1) Hardware status window

(2) Event notification

(3) Popup notification

(4) Outputs on the status display digital LEDs

(5) Remote notification

(6) Automatic shutdown

(7) Alarm lamp

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

For details about (6), see "5.1　Automatic Shutdown of the Equipment".

If you use the MCALL contact of the RAS external contacts interface, the external hardware connected to the RAS external contacts interface can detect malfunction of this equipment. The following table shows how the state of the MCALL contact changes.

Table 2-1　State of the Equipment in Terms of Fan Monitoring and State of the MCALL Contact

Status of this equipment	MCALL contact
During startup of the equipment or when the power is off	Open
When the fans operate normally or have recovered from malfunction	Open
During malfunction of the fans	Closed

The alarm lamp (as well as the MCALL contact of the RAS external contacts interface) is not dedicated to fan monitoring, and therefore, it is not turned off (opened) when the system recovers from fan malfunction as long as malfunction is detected for other monitored items.

2.2   Monitoring Temperature inside the Chassis

This function monitors the temperature inside the chassis using the temperature sensor in this equipment and notifies in the following methods when the temperature inside the chassis gets abnormally high.

(1) Hardware status window

(2) Event notification

(3) Popup notification

(4) Outputs on the status display digital LEDs

(5) Remote notification

(6) Automatic shutdown

(7) Alarm lamp

For details about (1) to (5), see "CHAPTER 4   CHECKING THE HARDWARE STATUS". For details about (6), see "5.1   Automatic Shutdown of the Equipment".

If you use the MCALL contact of the RAS external contacts interface, the external hardware connected to the RAS external contacts interface can detect malfunction of this equipment. The following table shows how the state of the MCALL contact changes.

Table 2-2   State of the Equipment in Terms of Monitoring Temperature inside the Chassis and State of the MCALL Contact

Status of this equipment	MCALL contact
During startup of the equipment or when the power is off	Open
When the temperature is normal or has recovered from being abnormally high	Open
When the temperature is abnormally high	Closed

The alarm lamp (as well as the MCALL contact of the RAS external contacts interface) is not dedicated to monitoring the temperature inside the chassis, and therefore, it is not turned off (opened) when the system recovers from the abnormally high temperature as long as malfunction is detected for other monitored items.

2.3   Drive Failure Prediction Function (SMART Monitoring)

The hard drives in this equipment have the Self-Monitoring, Analysis and Reporting Technology (SMART) function, which constantly monitors the condition of the drives and anticipates a failure before the failure manifests itself. The drive failure prediction function notifies in the following methods when there is a possibility that failure may occur in a drive in near future.

(1) Hardware status window

(2) Event notification

(3) Popup notification

(4) Outputs on the status display digital LEDs

(5) Remote notification

(6) The hfwDiskStat and GetHddPredict functions in the RAS library

For details about (1) to (5), see "CHAPTER 4   CHECKING THE HARDWARE STATUS". For details about (6), see "6.1.8   Get function for the failure prediction status of the drive (GetHddPredict)", "6.1.9   Get function for the drive condition (hfwDiskStat)".

NOTICE

When failure of a drive is anticipated, the drive may experience hardware failure in near future. We recommend you to back up the data and replace the drive. For information about how to replace a drive, refer to "HF-W2000 Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".

── NOTE ──────────────

- It is not possible for the Drive Failure Prediction Function to anticipate all failures. Therefore, a drive may fail before the Drive Failure Prediction Function anticipates any failures.
- This function only monitors the internal drives that are recognized during OS startup. If you connect a new drive or replace a drive with a new drive, for example, for maintenance, it may take a long time to recognize the new drive at the first startup after the new drive is connected, and the drive may not be recognized as a drive to be monitored. If this situation happens, restart this equipment.

2.4   Drive Usage Monitoring

The drive usage monitoring function adds up the power-on hours of a drive in this equipment. If the total power-on hours exceeds the set value, this function notifies in the methods below. By using this function, you can keep track of the appropriate timing of replacing a drive and prevent drive failure caused by using the drive for too long.

(1) Hardware status window

(2) Event notification

(3) Popup notification

(4) Remote notification

(5) The hfwDiskStat function in the RAS library

For details about (1) to (4), see "CHAPTER 4   CHECKING THE HARDWARE STATUS". For details about (5), see "6.1.9   Get function for the drive condition (hfwDiskStat)". This function can be enabled or disabled in the RAS Setup window. If this function is disabled, notification using the methods above is disabled. For details, see "3.1.3   Using the RAS Setup window".

NOTE

- For the S model, the default factory-shipped settings will be disabled. If you want to enable this function, change the notification threshold depending on the type of the mounted drive. For information about the setting of notification threshold, see "3.1.3   Using the RAS Setup window".
- This monitoring function is active while the OS is running, that is, from OS startup to OS shutdown. When the OS is not running, the amount of the drive power-on hours is not monitored.
- This function records the serial number (unique ID) of a drive in the equipment in a drive management information file. If you mount a drive that does not match the serial number recorded in the drive management information file, the cumulative power-on hours for the drive is automatically reset.
- This function is not designed to anticipate drive failure, but we recommend replacing a drive when the power-on hours exceeds the set value from the viewpoint of preventive maintenance for components with limited life span. For information about how to replace a drive, refer to "HF-W2000 Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".
- This function only monitors the internal drives that are recognized during OS startup. If you connect a new drive or replace a drive with a new drive, for example, for maintenance, it may take a long time to recognize the new drive at the first startup after the new drive is connected, and the drive may not be recognized as a drive to be monitored. If this situation happens, restart this equipment.

2.5 Memory Monitoring

Memory with Error Checking and Correcting (ECC) is installed in this equipment. A single bit error in the memory can be automatically corrected without causing any problems to the operation of the equipment. On the other hand, if the frequency of memory error correction is high, that is considered to be caused by memory failure, and we recommend replacing the memory modules from the viewpoint of preventive maintenance.

The memory monitoring function notifies in the following methods when the frequency of memory error correction is high.

(1) Event notification

(2) Popup notification

(3) Remote notification

(4) The GetMemStatus function in the RAS library

For details about (1) to (3), see "CHAPTER 4　CHECKING THE HARDWARE STATUS".

For details about (4), see "6.1.7　Get function for the memory condition (GetMemStatus)".

　　　　NOTE

For information about how to replace a memory module, refer to "HF-W2000 Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".

2.6 OS Hung-up Monitoring

The OS hung-up monitoring function uses a timer implemented on this equipment for monitoring the operational state of the OS (OS monitoring timer), in order to detect a situation that a process with the highest priority (real-time priority class) cannot run (hereinafter referred to as "OS hung-up") due to, for example, runaway kernel or when all CPU load is used up by a driver. When OS hung-up is detected, this function notifies it using the status lamp on the front of this equipment.

Table 2-3   State of the Equipment in Terms of OS Hung-up Monitoring and State of the Status Lamp

Status of this equipment	Status lamp
During startup of the equipment or when the power is off	Red
When OS hung-up has not occurred or the system has recovered from OS hung-up	Green
When OS hung-up has occurred	Red

If you use the CPUSTOP contact of the RAS external contacts interface, an external hardware connected to the RAS external contacts interface can detect OS hung-up. The following table shows how the state of the CPUSTOP contact changes.

Table 2-4   State of the Equipment in Terms of OS Hung-up Monitoring and State of the CPUSTOP Contact

Status of this equipment	CPUSTOP contact
During startup of the equipment or when the power is off	Closed
When OS hung-up has not occurred or the system has recovered from OS hung-up	Open
When OS hung-up has occurred	Closed

NOTICE

When OS hung-up occurs, processes on the OS cannot run as scheduled, and the facility using this equipment may be affected due to delay in processing. If OS hung-up occurs, resolve it immediately.

— NOTE

- For this function, OS hung-up is defined as a state that a process with the highest priority (real-time priority class) cannot run.
- When the OS starts, the status lamp is lit green at the timing when this function starts.
- When the OS shuts down, the status lamp becomes red when this function stops. It should be noted that when the status lamp turned red, the shutdown process is not complete. Do not turn off the power at this point.

Figure 2-1 shows how the state of the status lamp and the state of the CPUSTOP contact change.
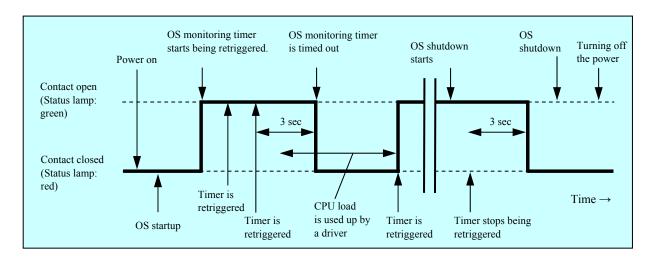
Figure 2-1   State of the Status Lamp and the CPUSTOP Contact

Dotted lines in Figure 2-1 denote the respective state of the contact and the thick line denotes the transition of the state of the contact.

2.7 Watchdog Timer Monitoring

This equipment has a watchdog timer. By retriggering a watchdog timer automatically, this function monitors whether processes are scheduled properly. In addition, this function can be used, for example, for monitoring the operational state of a user program by using dedicated library functions.

2.7.1 Automatic retriggering feature for a watchdog timer

This feature is implemented on a process retriggering a watchdog timer periodically. This process runs at the lowest priority (idle priority class). If the amount of CPU time used by processes with a priority higher than the idle priority class exceeds the set value, the watchdog timer is timed out. By using this feature, you can detect a runaway application process and so on.

You can set up the watchdog timer timeout and the retrigger interval in the RAS Setup window. For information about how to use the RAS Setup window, see "3.1.3 Using the RAS Setup window". The default factory settings are as follows.

Timeout	60 (seconds)
Retrigger interval	20 (seconds)

2.7.2   Using a watchdog timer for monitoring a user program

In order to use a watchdog timer for monitoring the operational state of a user program, you can use a configuration, for example, where the user program periodically retriggers the watchdog timer (that is, resets the timeout counter of the watchdog timer to the initial value) and another program checks whether timeout occurs for the watchdog timer. The flow chart for this configuration is shown below.
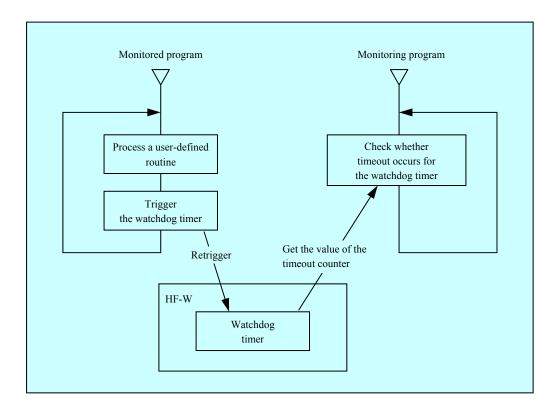
Figure 2-2   Example of a Flow Chart of Monitoring the Operational State of a User Program

In Figure 2-2, the monitored program is configured to periodically retrigger the watchdog timer.

The monitoring program periodically checks the timeout counter of the watchdog timer and if the value of the timeout counter (that is, the remaining time until timeout expires) is 0, it is determined that the timeout occurred.

The fact that timeout occurred means that the monitored program was not able to retrigger the watchdog timer for at least the length of the timeout. In this example, another program detects timeout of the watchdog timer, but if you use the RAS external contacts interface, the WDTTO contact is closed when timeout occurs for the watchdog timer. In this case, an external hardware connected to the RAS external contacts interface can monitor the operational state of a user program.

A program can use a watchdog timer by calling the library function WdtControl. For information about how to use the WdtControl function and how the WDTTO contact of the RAS external contacts interface works, see "6.1.3   Watchdog timer control function (WdtControl)".

───── NOTE ─────────────────────────────

When the WdtControl function is used, you cannot use the Automatic retriggering feature for a watchdog timer. In order to use the WdtControl function, select **Retriggered by application program** under **Watchdog timer setting**. For information about how to use the RAS Setup window, see "3.1.3 Using the RAS Setup window".

───

2.8 RAID Monitoring [B Model only]

B Model has a RAID1 feature (hereinafter denoted simply as "RAID") generally known as "mirror disk (mirroring)".

The RAID monitoring function monitors the status of the RAID on this equipment. If there is any change in the status of the RAID, this function notifies a user or an application using the following methods.

(1) Hardware status window

(2) Event notification

(3) Popup notification

(4) Outputs on the status display digital LEDs

(5) Remote notification

(6) Alarm lamp

(7) The hfwDiskStat, hfwRaidStat, and RaidStat functions in the RAS library

(8) RAID configuration control command (raidctrl)

For details about (1) to (5), see "CHAPTER 4 CHECKING THE HARDWARE STATUS". For details about (7), see "6.1.9 Get function for the drive condition (hfwDiskStat)", "6.1.10 Get function for the RAID status (hfwRaidStat, RaidStat) [B Model only]". For details about (8), see "5.3 RAID Configuration Control Command (raidctrl) [B Model only]".

If you use the MCALL contact of the RAS external contact interface, the external hardware connected to the RAS external contact interface can detect malfunctions of this equipment. The following table shows how the state of the MCALL contact changes.

Table 2-5 Status of the RAID and State of the MCALL Contact

RAID status	MCALL contact
Normal	Open
Degraded(*), Unknown	Closed

(*) This is also applicable when media errors are configured to be reported.

The alarm indicator (as well as the MCALL contact of the RAS external contact interface) is not dedicated just to monitoring the status of the RAID. Therefore, it will not be turned off (opened) when the status of the RAID recovers as long as malfunctions are detected in other monitored items.

2.8.1　State transition of the RAID

Table 2-6 lists RAID statuses and their respective descriptions. Figure 2-3 shows the state transitions.

Table 2-6　RAID Statuses and Their Descriptions

RAID status	Detailed information	Description
Normal	-	Redundant data is intact, and the RAID is operating normally.
Degraded	-	One drive failed, and a part of redundant data is corrupted.
Degraded	Rebuild	The RAID is being rebuilt due to drive replacement or other reason.
Fail	-	Both drives failed.
Unknown	-	Getting the RAID status failed, or the array configuration is not normal.
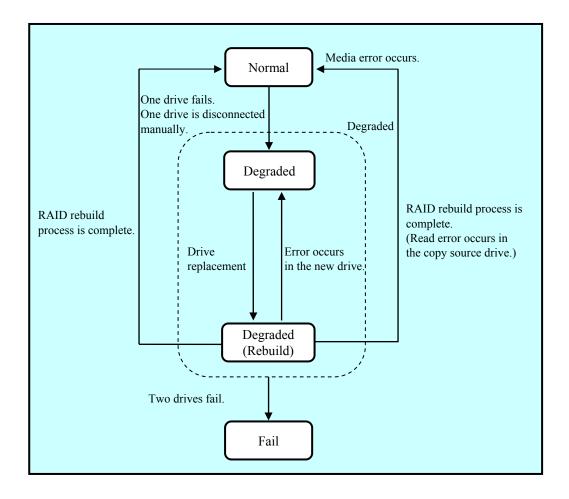
Figure 2-3　RAID Status Transitions

2.8.2   Note about media error

Media error is a status where there is a problem in data integrity while the RAID status is normal. If read errors occur at the copy source drive during a rebuild process, the rebuild process will complete but the sector data that could not be read is already lost, indicating a data integrity problem.

<Handling a media error as a failure>

The severity of this data integrity problem depends on whether those bad sectors with read errors are actually used. If those bad sectors are used, there is a real problem in data integrity. But if not, that is not a problem at all.

This is why this equipment allows you, depending on how the system is operated, to select whether a media error is reported. In the default factory settings, a media error is not reported.

Use the RAID configuration control command (raidctrl) to select whether a media error is reported. (For details, see "5.3   RAID Configuration Control Command (raidctrl) [B Model only]".)

A media error can be reported in the following ways:

(1) Event notification

(2) Outputs on the status display digital LEDs

(3) Remote notification

(4) Alarm lamp

(5) Message box

(6) MCALL contact of the RAS external contacts interface
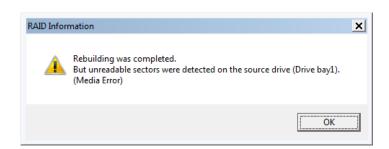
Figure 2-4   Example of a Message Box when a Media Error Occurs

Note that, regardless of the setting for whether a media error is reported, you can check whether media errors occurred in the following ways:

(7) Hardware status window

(8) The hfwRaidStat function in the RAS library

(9) RAID configuration control command (raidctrl)

For details about (1) to (3) and (7), see "CHAPTER 4　CHECKING THE HARDWARE STATUS". For details about (8), see "6.1.10　Get function for the RAID status (hfwRaidStat, RaidStat) [B Model only]". For details about (9), see "5.3　RAID Configuration Control Command (raidctrl) [B Model only]".

CHAPTER 3 SETTING UP THE RAS FEATURE

3.1 RAS Setup Windows

3.1.1 Overview

In the **RAS Setup** window, the following functions can be set up.

Table 3-1   Setup Items in the RAS Setup Window

Item	
Shutdown setting	Automatic shutdown when detecting fan failure
	Automatic shutdown when detecting abnormally high temperature
	Automatic shutdown when detecting remote shutdown request
Watchdog timer setting	
Drive usage monitoring setting	
Status display digital LEDs setting	
Popup notification setting	

Figure 3-1 shows the **RAS Setup** window. The default factory-shipped settings are shown in the figure.
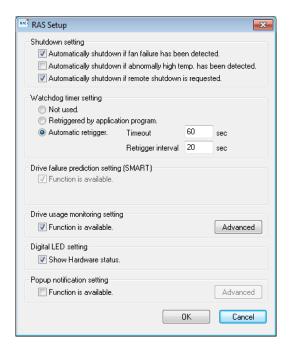
Figure 3-1   RAS Setup Window

3.1.2   Starting the RAS Setup window
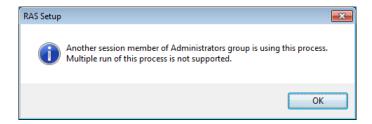
To start the **RAS Setup** window, follow the procedure below.

Before you start this window, you need to log on to the computer as an administrator account.

1. Click **Start**.

2. Point to **All Programs** (or **All Apps**) > **RAS Software**.

3. Click **RAS Setup**.

NOTE

The **RAS Setup** window cannot be used by multiple users at the same time. If you use, for example, user switching to try to start instances of this window from multiple consoles, the following message appears. If you receive this message, close the **RAS Setup** window on another console, and then start the **RAS Setup** window.

RAS Setup

Another session member of Administrators group is using this process. Multiple run of this process is not supported.

OK

3.1.3   Using the RAS Setup window

(1) Shutdown setting

You can select whether this equipment is automatically shut down for each of the following cases: a fan failure, an abnormally high temperature, and a remote shutdown request.
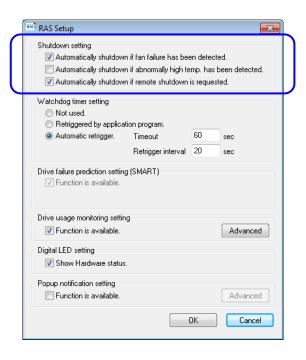
Figure 3-2   Items in the Shutdown Setting

● "**Automatically shutdown if fan failure has been detected**" check box
  • Selected: This equipment will be automatically shutdown (default factory-shipped setting).
  • Cleared: This equipment will not be automatically shutdown.

● "**Automatically shutdown if abnormally high temp. has been detected"** check box
  • Selected: This equipment will be automatically shutdown.
  • Cleared: This equipment will not be automatically shutdown (default factory-shipped setting).

● **"Automatically shutdown if remote shutdown is requested"** check box
  • Selected: This equipment will be automatically shutdown (default factory-shipped setting).
  • Cleared: This equipment will not be automatically shutdown.

If you change the current setting, click the check box you want to change.

NOTICE

If this equipment continues to operate while fan failure is detected, internal parts such as a processor are not cooled down sufficiently, and that may cause thermal runaway of the system due to malfunction of the equipment or result in damage to the parts.

NOTE

- If you clear the **"Automatically shutdown if fan failure has been detected"** check box, the following cautionary message is displayed. Enable this function (select the check box) whenever you can.

If you click **Cancel** when you receive the message above, the check box reverts back to being selected. If you click **OK**, the check box is cleared.
- After an automatic shutdown initiated by this function is complete, the power is turned off.
- The remote shutdown input is polled every 5 seconds. That means it may take up to 5 seconds from when a remote shutdown is requested to when an actual shutdown process starts.

---

(2) Watchdog timer setting

You can set up the watchdog timer of this equipment.

You can select one of the following ways of using the watchdog timer:

• Not used

• Retriggered by an application program

• Automatic retrigger

Click the radio button corresponding to the item you want to select.

The default factory setting is "**Not used**".
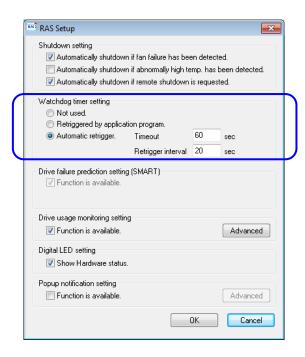
Figure 3-3   Items in Watchdog Timer Setting

● **Not used:**

If you select this item, the watchdog timer is stopped. The watchdog timer will not time out. In addition, you cannot use the watchdog timer by calling the WdtControl function of the RAS library.

● **Retriggered by application programs:**

If you select this item, you can monitor the operational state of a user program by controlling the watchdog timer using the WdtControl function in the RAS library.

――― NOTE ―――

When you change the setting to "**Retriggered by application program**," the watchdog timer is stopped once. While the watchdog timer is stopped, it will never time out. When a user application retriggers the watchdog timer by using the WdtControl function of the RAS library, the watchdog timer resumes a countdown. The state of the watchdog timer (whether it is counting down or stopped) can be checked using the WdtControl function.

● **Automatic retrigger:**
If you select this item, the watchdog timer is automatically retriggered by the automatic retriggering feature for the watchdog timer. In addition, you cannot use the watchdog timer by calling the WdtControl function of the RAS library.

If this item is selected, you can configure the retrigger interval of the automatic retriggering feature and the timeout of the watchdog timer.

• Timeout
Sets the timeout of the watchdog timer by using single-byte numeric characters. Type an integer value between 5 and 60 (in seconds). The default factory-shipped setting is 60 (seconds).

• Retrigger interval
Sets the retrigger interval of the automatic retriggering feature for the watchdog timer by using single-byte numeric characters. Type an integer value between 1 and (timeout - 4) (in seconds). The default factory-shipped setting is 20 (seconds).

You can set a timeout and retrigger interval only if you select **"Automatic retrigger."** If **"Automatic retrigger"** is not selected, you cannot enter those values.

(3) Drive usage monitoring setting
You can set up the drive usage monitoring setting. By clicking **Advanced**, you can set up the advanced settings of this function.
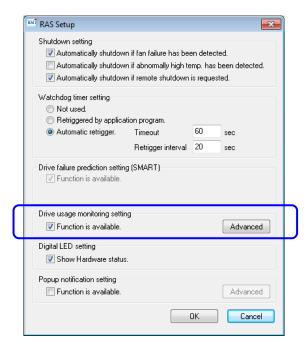
Figure 3-4   Items in the Drive Usage Monitoring Setting

● **"Function is available"** check box
• Selected: The drive usage monitoring is enabled.
        (default factory-shipped setting for A/B model)
• Cleared: The drive usage monitoring is disabled.
        (default factory-shipped setting for S model)

If you change the current setting, click the check box.
If the drive usage monitoring is disabled, you cannot use the notification functions described in "2.4   Drive Usage Monitoring" such as indicating if the value of the drive power-on hours exceeds the threshold.

── NOTE ─────

For the S model, the default factory-shipped settings will be disabled. If you want to enable this function, change the notification threshold depending on the type of the mounted drive.

● **Advanced** button
If the drive usage monitoring is enabled, click **Advanced** to display the following
window. If the drive usage monitoring is disabled, this item cannot be set.
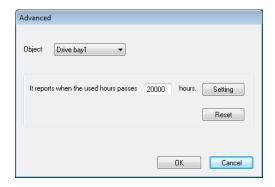
Figure 3-5   Advanced Setting for the Drive Usage Monitoring

In the **Advanced** window, you can set up a threshold for the drive power-on hours so
that you can be notified when the value of the drive power-on hours exceeds the
threshold. To set the notification threshold, from the **Object** list, select the drive you
want to set up. Type the value for the threshold in single-byte characters in the box in
"It reports when the used hours passes ☐ **hours."** Then click **OK**.

Enter an integer value in hours ranging from 100 to maximum value in the step of 100.
The default factory-shipped setting is 20000 (hours).
The maximum value that can be set is the following values.
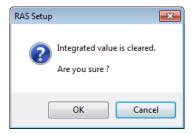
A/B model:	20000 (hours)
S model:	99900 (hours)

─── NOTE ───
If you want to enable this function in the S model, set the notification threshold
depending on the type of the mounted drive. For the recommended
replacement cycle of each drive, refer to the "HF-W2000 Model 48/45
INSTRUCTION MANUAL (manual number WIN-62-0066)".
If the type of the mounted drive is an HDD, it is recommended that you set to
20,000 hours from the point of view of the replacement cycle.

If you want to clear the current cumulative power-on hours when, click **Reset** to clear the value. Then the following message is displayed.

If you click **OK** when you receive the message, the value of the current cumulative power-on hours is cleared. Click **Cancel** to leave the value as is.

If you changed the settings in the **Advanced** window and want to make the change effective, click **OK**.
If you do not want the change, click **Cancel**. The window is closed and the change is discarded.

(4) Status display digital LEDs setting

You can set up the display mode of the status display digital LEDs located on the front of the equipment.
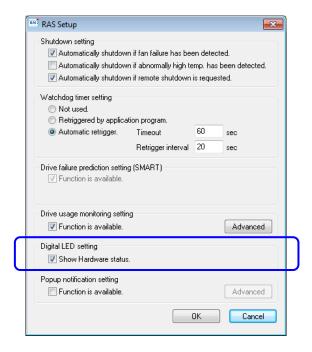
Figure 3-6   Items in the Status Display Digital LEDs Setting

● **"Show Hardware status"** check box
  • Selected: Hardware status display mode (default factory-shipped setting).
  • Cleared: Application status display mode.

If you change the current setting, click the check box.
For details about the function of the status display digital LEDs including its display modes, see "4.4   Status Display Digital LEDs Function".

(5) Popup notification setting

You can set up the popup notification setting. By clicking **Advanced**, you can set up the advanced settings of this function.
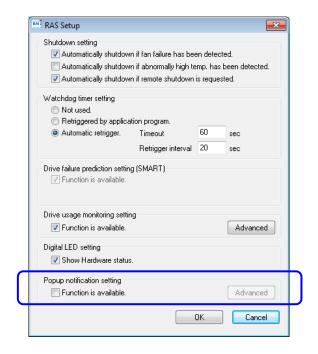
Figure 3-7   Items in the Popup Notification Setting

● **"Function is available"** check box
   • Selected: The popup notification is enabled.
   • Cleared: The popup notification is disabled (default factory-shipped setting).

If you change the current setting, click the check box.
If the popup notification is disabled, you cannot use the popup notification feature described in "4.3   Popup Notification".

● **Advanced** button
Click **Advanced** to display the following window.

| A/S Model | B Model |
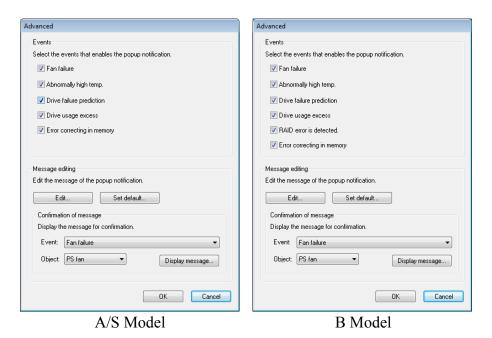
Figure 3-8   Advanced Settings for the Popup Notification Setting

[Events]
  • Fan failure
  • Abnormally high temp.
  • Drive failure prediction (SMART)
  • Drive usage excess (When the drive power-on (=used) hours exceeds the threshold)
  • RAID failure detection (B Model only)
  • Error correcting in memory (When memory correction is detected frequently)
    You can disable or enable popup notification for each of the items above.

● Check box for each item
  • Selected: The popup notification is enabled. (factory-shipped default setting)
  • Cleared: The popup notification is disabled.
    If you change the current setting, click the check box.
    If the monitoring function is disabled, it will not be notified even if the popup notification is enabled.

[Message editing]

You can edit popup notification messages. And you can check the messages after you edit them. For information about how to edit and check the messages, see "3.1.4   Editing popup notification messages".

If you changed the advanced setting and want to make the change effective, click **OK**. If you do not want the change, click **Cancel**. Then the window is closed and the change is inapplicable.

(6) Making the change you made in (1) through (5) effective or discard the changes

If you changed the settings in (1) through (5) and want to make the change effective, click **OK**. The **RAS Setup** window is closed and the settings become effective immediately.

If you do not want the change the setting, click **Cancel**. Then the **RAS Setup** window is closed and the changes are discarded.

3.1.4   Editing popup notification messages
(1) Editing popup notification messages
If you want to edit messages used for popup notification, click **Edit**. Notepad is launched and the message definition file for popup notification is opened. Edit the messages in the format described below.
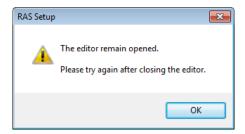
────── NOTE ──────

While you are editing messages, you cannot do the following:
• Clicking **Edit**.
• Clicking **Set default**.
• Clicking **Display message**.
• Closing the **RAS Setup** window (clicking **OK** or **Cancel**).
If you do one of the above, the following cautionary message is displayed.

If you click **OK** when you receive the message, you go back to the **RAS Setup** window.

■ Format of a definition file
The format of a definition file is as follows.
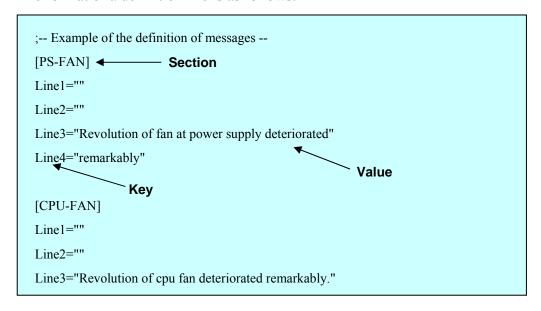
Figure 3-9   Format of a Definition File

A definition file consists of sections, keys, and the values of the keys.
Each section contains keys and their values. A key and its value are separated by an equal sign (=).
Lines that start with a semicolon (;) are comment lines.

■ Description in a definition file

1. Section

The table below shows a list of section names you can define for this function and an explanation of the message you define for each section.

Table 3-2   Section Names and Defined Messages

Section name	Defined message
[PS-FAN]	A message displayed when a PS fan failure is detected.
[SYS-FAN]	A message displayed when a System fan failure is detected.
[CPU-FAN]	A message displayed when a CPU fan failure is detected.
[TEMP]	A message displayed when an abnormal temperature inside the chassis is detected.
[HDD1-SMART] (*1)	A message displayed when a drive failure prediction (SMART) is detected for the drive bay1.
[HDD2-SMART] (*1)	A message displayed when a drive failure prediction (SMART) is detected for the drive bay2.
[HDD1-OFFLINE] (*2)	A message displayed when a RAID failure is detected due to a drive failure at drive bay1.
[HDD2-OFFLINE] (*2)	A message displayed when a RAID failure is detected due to a drive failure at drive bay2.
[HDD1-OVERRUN] (*1)	A message displayed when the drive power-on (=used) hours exceeds the determined value for the drive bay1.
[HDD2-OVERRUN] (*1)	A message displayed when the drive power-on (=used) hours exceeds the determined value for the drive bay2.
[DIMM1-ERR]	A message displayed when an error correction is detected frequently in DIMM A1.
[DIMM2-ERR]	A message displayed when an error correction is detected frequently in DIMM B1.

(*1) Regardless of the mounted drive, this section name is fixed.
(*2) B Model only.

2. Keys

For a key, specify the line number of a line displayed as a part of the popup message.

In this function, you can use the keys "Line1" through "Line5" for each section.

If you specify Line6 and so on for keys, those keys are ignored.

3. Values

Specify one line of message displayed as a part of the popup message for a value.
For each key, a maximum of 50 bytes of characters (up to 25 double-byte characters)
can be assigned. If you specify more than 50 bytes of characters, the characters at the
51th byte and after are ignored.
If the line includes space characters, the whole value should be enclosed in double
quotation marks (""). If the value is empty, it is treated as a newline character.

───── NOTE ─────

- When you save the change, make sure you use **Save** in the menu. Otherwise,
  the change you made may not be saved properly.
- While you are editing a definition file, do not use another application to edit
  the same file. If you edit a definition file from multiple applications at the same
  time, the change may not be saved properly.
- When you edit a popup notification message, make sure that the message
  clearly states that an error occurred. If you continue to use this equipment
  while error remains, your system may be affected significantly.
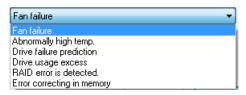
(2) Checking popup notification messages

You can check the change you made in the message for each of the following items:

• Fan failure

• Abnormally high temp.

• Drive failure prediction

• Drive usage excess (When the drive power-on (=used) hours exceeds the threshold)

• RAID error is detected. (B Model only)

• Error correcting in memory (When memory correction is detected frequently)

The following shows how to check the change you made in the message.

1. In the Event list, select the event you want to check.

In this list, only the items under **Events** selected using the check boxes are displayed. If no items under **Events** are selected using the check boxes, you cannot select an item from the **Event** list.

<Example of Selecting Fan failure>

2. In the **Object** list, select the object you want to check.

Items in this list depend on the item you selected in step 1.

<Example of selecting PS (power) fan>
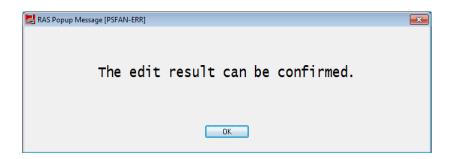
The table below shows items displayed in the **Object** list for each option you select in the **Event** list.

Table 3-3 Items Displayed in the Object List for Each Option Selected in the Event List

Option in the Event list	Items in the Object list
Fan failure	PS fan, System fan, CPU fan
Abnormally high temp.	Temperature inside the chassis
Drive failure prediction	Drive bay1, Drive bay2
Drive usage excess	
RAID error is detected.	Drive bay1, Drive bay2
Error correcting in memory	DIMM A1, DIMM B1

3. Click **Display message**. A popup notification message is displayed based on the change you made. After you confirm that the message is OK, click **OK** in the popup.
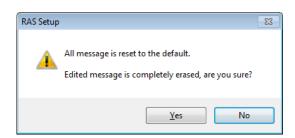
If the message is not edited or there is something wrong in the message definition file, the following message is displayed. Click **OK** to go back to the **RAS Setup** window. Correct the change you made in the message.

(3) Restoring default messages

If you want to set the popup notification messages back to the default, click **Set default**. The following message will appear. Click **Yes**. Then the changes you made in the message definition file are discarded.

If you click **No**, the changes are not deleted and the messages do not revert back to the default.

CHAPTER 4 CHECKING THE HARDWARE STATUS

You can check the hardware status of this equipment by using the following methods.

(1) Check by using GUIs

You can check the hardware status of this equipment by using a graphical interface. For details, see "4.1　Hardware Status Window".

(2) Check with a user application

A user application can check the hardware status of this equipment by monitoring the status of event objects. For details, see "4.2　RAS Event Notification".
A user application can also get the hardware status of this equipment by using the RAS library. For details, see "4.6　Status Acquisition by Using the RAS Library".

(3) Check on the desktop of this equipment

A popup message is displayed to notify that an error occurred in the hardware of this equipment. For details, see "4.3　Popup Notification".

(4) Check by using the status display digital LEDs on the front of this equipment

The status display digital LEDs are on the front of this equipment and notify a user that an error has occurred in the hardware of this equipment. These LEDs can also be used by a user application in order to, for example, notify maintenance personnel of the failure. For details, see "4.4　Status Display Digital LEDs Function".

(5) Check from a remote device

A remote device can check the hardware status of this equipment. A remote device can also be notified whenever the hardware status changes. For details, see "4.5　Remote Notification".

4.1 Hardware Status Window

4.1.1 Overview

After you log on to this equipment, there will always be an icon in the notification area of the taskbar to display the hardware status. If you double-click this icon or if you right-click the icon to display a popup menu and click **Display Hardware status**, detailed information about the hardware status of this equipment is displayed.

This window displays the following information:

• Fan condition
• Temperature condition inside the chassis
• Drive failure prediction (SMART monitoring) condition
• Drive power-on (=used) hours (In the default factory-shipped setting of the S model, this information is not displayed)
• RAID status (B Model only)

A/S Model   B Model
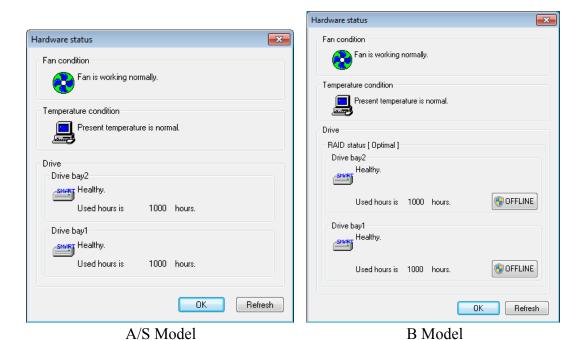
Figure 4-1   Hardware Status Window

NOTICE

When the **Hardware status** window shows an error in hardware, resolve the error immediately.

—— NOTE ——

- For information about how to replace a perishable component, refer to "HF-W 2000 Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".
- This window only displays the internal drives that are recognized during OS startup. If you connect a new drive or replace a drive with a new drive, for example, for maintenance, it may take a long time to recognize the new drive at the first startup after the new drive is connected, and the information about the drive may not be displayed. If this situation happens, restart this equipment.

4.1.2   Hardware status icon

There will always be an icon in the notification area of the taskbar to display the hardware status.

Note that, in the default factory settings, the icon is not shown. If you click the arrow at the side of the notification area, the icon will appear. If you want to always show the icon in the notification area of the taskbar, click **Customize** and configure the icon to be always shown in the notification area. (If you are using Windows 10, right-click the arrow at the side of the notification area. A menu will appear. On the menu, click **Properties**. Then, a window will appear. In the window, click **Customize**, and then **Select which icons appear on the taskbar**. Then, configure the icon to be always shown in the notification area in the taskbar.)

──── NOTE ────

In rare cases, the registration of the hardware status icon fails and the following message is displayed. If this happens, follow the procedure below to retry the registration of the hardware status icon.

1. Click **OK** in the message above.
2. Point to **Start** > **All Programs**(or **All Apps**) > **Startup**(or **RAS Software**), and click **RAS Status**.

(1) List of displayed icons and description of each icon

Table 4-1 shows a list of displayed icons and a description of each icon. A description of the displayed icon is shown when you point the mouse cursor to the icon.

Table 4-1　Hardware Status Icon

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

Nos. 5 through 7 and 11: When both "Time of drive exceed." and "Drive failure possible." happen at the same time, the description of the icon does not show "Time of drive exceeded."

Nos. 11 through 14: These descriptions apply to B Model only. When both "Drive failure possible." (or "Time of drive exceeded.") and disconnection of a drive due to RAID error (OFFLINE) happen at the same time, the description of the icon does not show "Drive failure possible." (or "Time of drive exceeded.").

Nos. 15 and 16: If a hardware status error is detected at the same time, the icon for a hardware status error is displayed.

Figures 4-2 and 4-3 show examples of displaying the description of an icon when the hardware status of this equipment is normal and when the hardware status has an error.

Figure 4-2   Example of Displaying the Description of an Icon (When the Hardware Status Is Normal)
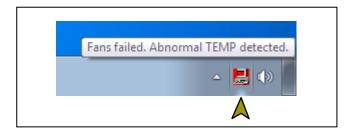
Figure 4-3   Example of Displaying the Description of an Icon (When the Hardware Status Has an Error)

(2) Menu of the hardware status icon

If you right-click the icon, a popup menu is displayed.

Right-click the icon to display a popup menu.
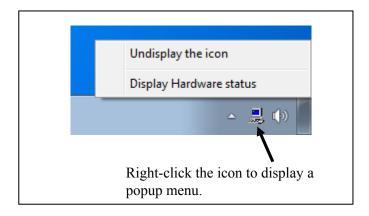
Figure 4-4   Menu of the Hardware Status Icon

● **Display Hardware status**

Click it to display the **Hardware status** window.

● **Undisplay the icon**

Click it to delete the icon from the notification area of the taskbar.

4.1.3   Hardware status window

The **Hardware status** window shows the details of the hardware status of this equipment.
Figure 4-5 shows how to start the **Hardware status** window.
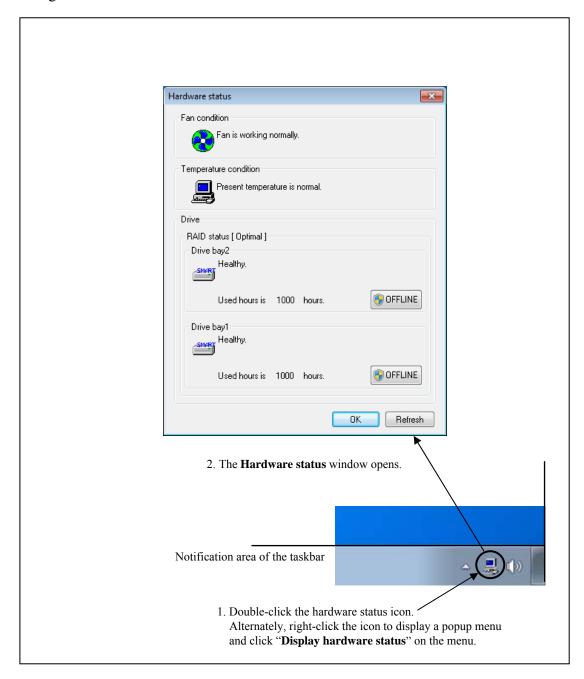
Figure 4-5   Starting the Hardware Status Window

(1) Description of the window

   1. Fan condition

     Shows the current status of the fans.

Table 4-2   Fan Condition and Displayed Information

Fan condition	Icon	Information
Normal		Fan is working normally.
Excessively low rotation speed		Fan failure is detected. For details, refer to the event log.

   2. Temperature condition

     Shows the current status of the temperature inside the chassis.

Table 4-3   Temperature Condition and Displayed Information

Temperature status	Icon	Information
Normal		Present temperature is normal.
Abnormally high temperature		Temperature in the unit has exceeded the upper limit.

3. Drive condition

Shows the current status of the drives. In the following areas, the drive conditions of drive bay1, 2 are displayed.

Table 4-4   Drive Condition and Displayed Information

No.	Drive condition		Icon	Information
1	Normal			Healthy.
2	Failure anticipation by SMART			A failure may be imminent.
3	Drive usage excess			Used hours exceeded prescribed value.
4	Error	The drive is offline.		OFFLINE.
5		The drive is being rebuilt.		REBUILD.
6		Unknown		Unknown.
7	Not mounted			Not Connected.

Nos. 4 through 6: These descriptions apply to B Model only.

NOTICE

When failure of a drive is anticipated, the drive may experience hardware failure in near future. We recommend you to back up the data and replace the drive. For information about how to replace a drive, refer to "HF-W2000 Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".

---

4. Drive used hours

The drive used hours (cumulative hours up to date) is displayed. The value of the cumulative hours is updated every hour between 0 and 100 hours and every ten hours after that. The range of the cumulative hours that can be displayed is between 0 and 99990 (hours). If the drive usage monitoring is disabled, the drive used hours cannot be displayed.

5. RAID status [B Model only]

The RAID status is displayed.

Table 4-5   RAID Status and Displayed Information

No.	RAID status	Detailed information	Displayed information	Note
1	Normal	-	[Optimal]	
2			[Optimal (Media Error)]	Media error occurs.
3	Degraded	-	[Degrade]	
4			[Degrade (Media Error)]	
5		Rebuild	[Degrade (Rebuild: xx%)]	xx: Progress of the rebuild process
6			[Degrade (Rebuild: xx%, Media Error)]	xx: Progress of the rebuild process Media error occurs.
7	Unknown	-	[Unknown]	

6. **OFFLINE** button [B Model only]

Click this button to disconnect the corresponding drive from the RAID.

This button works only when the drive condition of a drive in the RAID is either "Normal", "Failure anticipation by SMART", or "Drive usage excess".

In order to disconnect a drive, you must have administrator privileges. Log on to the computer using an administrator account and click **OFFLINE**.

If you click **OFFLINE**, a confirmation message similar to the one shown in Figure 4-6 will be displayed. To disconnect the drive from the RAID and set the drive to offline, click **Yes**. If you click **No**, the drive will not be disconnected.

Figure 4-6   Confirmation Message when Disconnecting the Drive in Drive Bay1

<hr>

NOTE

- A disconnected drive cannot be set back to online and used again. When you disconnect a drive, make sure that the correct drive is selected to be disconnected.
- Note that if "User Account Control" (UAC) is disabled and a user without administrator privileges clicks **OFFLINE**, the following message will be displayed and the drive will not be disconnected. When you need to disconnect a drive, log on to the computer using an administrator account and click **OFFLINE**.
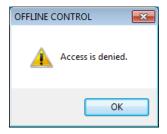
Figure 4-7   Message when a Request for Disconnecting the Drive in Drive Bay1 Is Rejected.

- You cannot disconnect a drive from a remote session.

<hr>

7. **Refresh** button

If you click this button, the latest hardware status is acquired and the information in the window is refreshed.

8. **OK** button

Click this button to close the **Hardware status** window.

Figure 4-8 shows the **Hardware status** window (B Model) when there is a hardware status error.

Figure 4-8   Hardware Status Window (Error Case)

4.2 RAS Event Notification

4.2.1 Overview

When an event that must be reported to a user such as hardware failure occurs, this function notifies the event to an application by setting an event object to the signaled state. An application can detect an event such as hardware failure by monitoring when the event object is set to the signaled state.

The event object is reset to the nonsignaled state when the cause of the event is cleared.

4.2.2 Detecting an event

Follow the procedure below to detect an event:

1. Use the OpenEvent Windows API function to get the handle to the event object. Specify SYNCHRONIZE for the parameter dwDesiredAccess (the access to the event object).

2. Use the WaitForSingleObject or WaitForMultipleObject Windows API function to monitor when the event object is set to the signaled state.

Table 4-6 is a list of events to be reported to a user and their respective event objects.

Table 4-6   Reported Events

No.	Event	Event object name
1	A PS fan failure occurred.	W2KRAS_PSFAN_ERR_EVENT
2	A System fan failure occurred.	W2KRAS_SYSFAN_ERR_EVENT
3	A CPU fan failure occurred.	W2KRAS_CPUFAN_ERR_EVENT
4	The temperature inside the chassis became abnormal.	W2KRAS_TEMP_ERR_EVENT
5	A remote shutdown request was generated.	W2KRAS_RMTSTDN_EVENT
6	SMART anticipated failure in one of the drives.	W2KRAS_HDD_PREDICT_EVENT
7	The power-on (=used) hours exceeded the threshold in one of the drives.	HFW_HDD_OVERRUN_EVENT
8	A frequent error correction occurredin one of the memory slots.	HFW_MEMORY_ERR_EVENT
9	The RAID status is normal.	HFW_RAID_OPTIMAL_EVENT
10	The RAID status is not normal.	HFW_RAID_DEGRADE_EVENT

No.6: This includes the case that getting the failure prediction status of the drive fails.
Nos. 9 and 10: Apply to B Model only.
No.10: This is also applicable when media errors are configured to be reported.

─── NOTE ───

When you use an event object in a program, you need to add "Global\" to the beginning of the name of the event object.

If the operating system is a 32-bit version, in addition to the objects described above, the following event objects are provided in order to maintain backward compatibility with the Windows NT® programs and old equipment.

Table 4-7   Reported Events (for Backward Compatibility with Old Programs)

No.	Event	Event object name
1	The temperature inside the chassis became abnormal.	SMGRASBD_TEMP_ERR_EVENT
2	A remote shutdown request was generated.	SMGRASBD_RMTSTDN_EVENT
3	SMART anticipated failure in one of the drives.	SMGSMART_HDD_PREDICT_EVENT
4	The power-on (=used) hours of the drive on the drive bay1 exceeded the threshold.	HFW_HDD1_OVERRUN_EVENT
5	The power-on (=used) hours of the drive on the drive bay2 exceeded the threshold.	HFW_HDD2_OVERRUN_EVENT
6	The RAID status is normal.	HFW_RAID_ONLINE_EVENT
7	The RAID status is not normal.	HFW_RAID_OFFLINE_EVENT

No.3: This includes the case that getting the failure prediction status of the drive fails.
Nos. 6 and 7: Apply to B Model only.
No.7: This is also applicable when media errors are configured to be reported.

—— NOTE ——

• When you use an event object in a program, you need to add "Global\" to the beginning of the name of the event object.
• The event objects listed in Table 4-7 are not provided if the operating system is a 64-bit version.
  If you are using a 64-bit operating system, use the event objects listed in Table 4-6.

4.2.3   Example of using event objects

We provide a sample program in C (FanErr.c) to show how to monitor event objects. For information about the name of the sample program and where you can find it, see "6.2 Sample Programs".

4.3   Popup Notification

4.3.1   Overview

When an event that must be reported to a user such as hardware failure occurred, this function notifies the event to a user by displaying a popup message on the desktop. Using this function, a user can know an event such as hardware failure occurred.

More specifically, a popup message is displayed in the following cases:

● A fan failure occurred.

● The temperature inside the chassis becomes abnormal.

● SMART anticipates drive failure.

● The drive power-on (=used) hours exceeded the threshold.

● A frequent error correction occurs in memory.

● A RAID error is detected. (B Model only)

Figure 4-9 shows an example of popup message notification when a power fan failure has occurred.
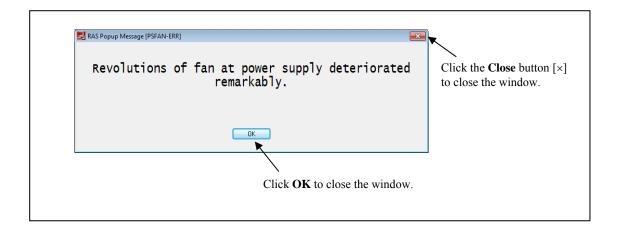
Figure 4-9   Example of Popup Message

4.3.2   Messages to be displayed

The following table shows a list of popup notification messages this function outputs. It should be noted that you can edit those messages. For information about how to edit messages, see "3.1.4   Editing popup notification messages".

Table 4-8   Messages Displayed

No.	Event	Popup notification message
1	A PS fan failure occurs.	Revolutions of fan at power supply deteriorated remarkably.
2	A System fan failure occurs.	Revolutions of System fan deteriorated remarkably.
3	A CPU fan failure occurs.	Revolutions of CPU fan deteriorated remarkably.
4	The temperature inside the chassis becomes abnormal.	Temperature exceeded prescribed value.
5	SMART anticipates drive failure.	A failure may be imminent on the drive of the drive bay%1.
6	The value of the drive power-on (=used) hours exceeds the threshold.	Used hours on the drive of the drive bay%1 exceeded prescribed value.
7	Frequent error correction occurs in memory.	In the %1, error corrections have occurred with high frequency.
8	A RAID error occurs.	RAID status is DEGRADE. (Drive bay%1)

Nos. 5, 6, and 8: %1 denotes the drive bay number.
No.5: This includes the case that getting the failure prediction status of the drive fails.
No.7: %1 denotes the name of DIMM.
No.8: Supported only by B Model only.

4.3.3   Popup notification settings

This function can be enabled or disabled in the **RAS Setup** window. In the default factory-shipped setting, this function is disabled. If this function is disabled, popup messages are not displayed.

For details, see "3.1.3   Using the RAS Setup window".

4.4   Status Display Digital LEDs Function

4.4.1   Overview

When an event that must be reported to a user such as hardware failure occurs, this function notifies the event to a user by displaying a code on the status display digital LEDs located on the front of this equipment. Using this function, a user can know an event such as hardware failure occurred.

In addition, a user application can output any code on the status display digital LEDs by using library functions. A code is displayed in a two-digit hexadecimal number.

The system status is displayed in two-digit hexadecimal code.

Status indication LEDs
①: RAS status
②: Application status
③: BIOS status
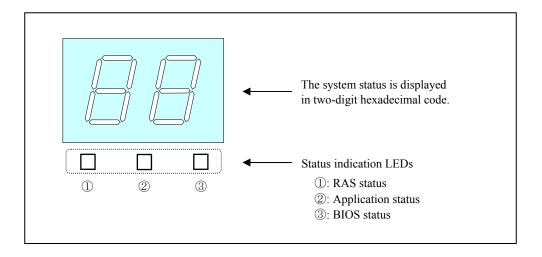
Figure 4-10   Status Display Digital LEDs

4.4.2   Status codes to be displayed
(1) Hardware status code

A hardware status code is displayed when an error has occurred in the hardware status of this equipment. If the hardware status is normal, a hardware status code is not displayed.

When a hardware status code is displayed, the leftmost LED in the status indication LEDs is lit.
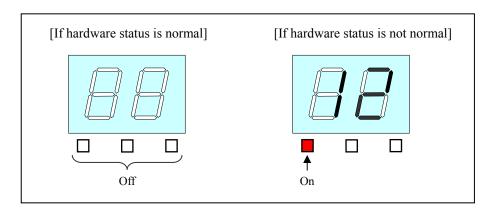
Figure 4-11   Hardware Status Codes

Table 4-9 shows a list of hardware status codes.

Table 4-9   Hardware Status Codes

No.	Status code	Cause	Priority
1	11	PS fan failure	
2	12	System fan failure	2
3	13	CPU fan failure	
4	21	The temperature inside the chassis becomes abnormal.	3
5	31	A failure is anticipated for the drive on drive bay1.	
6	32	A failure is anticipated for the drive on drive bay2.	4
7	41	Drive bay1 is offline.	
8	42	Drive bay2 is offline.	
9	4c	RAID is in an unknown status.	1
10	4d	Media error on RAID	

Nos. 7 through 10: Those status codes are displayed for B Model only.
No.10: This code is displayed if media error is configured to be reported as an error.

If multiple failures occur at the same time, the status code with the highest priority (code with the smallest value in the Priority column in Table 4-9) is displayed. If multiple failures with the same priority occur at the same time, the status code of the last detected failure is displayed.

(2) Application status codes

An application status code is displayed by a user application by using library functions provided by this function.

When an application status code is displayed, the center LED in the status indication LEDs is lit.
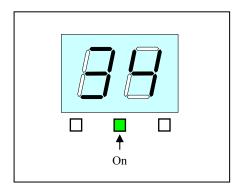
Figure 4-12   Application Status Code

(3) STOP error codes ("80" is displayed)

The STOP error code "80" is displayed when a STOP error (blue screen) has occurred during system operation for some reason. Regardless of the status display mode, "80" is displayed with higher priority than the timeout code. For information about the status display modes, see "4.4.3   Status display modes".

When the STOP error code is displayed, all LEDs in the status indication LEDs are lit.
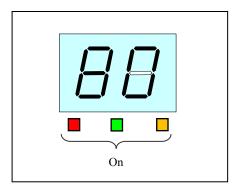
Figure 4-13   STOP Error Code

For information about the priority of displaying each code, see "4.4.4   Priorities of displayed codes".

4.4.3   Status display modes

This function has two display modes: "hardware status display mode" and "application status display mode".

Table 4-10   Status Display Modes

Status display mode	Description
Hardware status display mode	When the hardware status is normal, an application status code is displayed. When the hardware status has an error, a hardware status code has the higher priority to be displayed.
Application status display mode	Only an application code is displayed. Even when the hardware status has an error, the hardware status code is not displayed.

The status display mode can be configured in the **RAS Setup** window. The default factory-shipped setting is the hardware status display mode.

For information about how to use the **RAS Setup** window, see "3.1.3   Using the RAS Setup window".

Note that if the hardware status is normal and there is no application status code to display, the LEDs are all off.

Figure 4-14 shows an example of what is displayed in the hardware status display mode. Dotted lines in Figure 4-14 denote the respective display modes, and the thick line denotes the transition of the state of the LEDs.
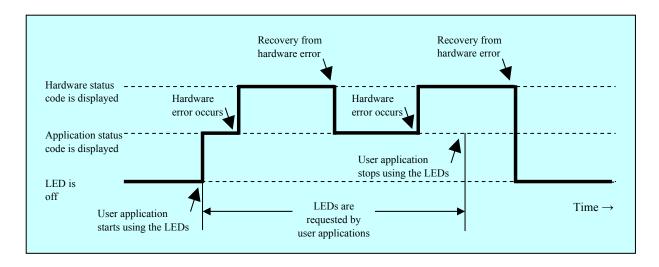
Figure 4-14   Example of a Status Display Mode

4.4.4   Priorities of displayed codes

The following shows the priorities of the codes displayed by this feature.

 (1) In the hardware status display mode

Code class	Priority order
STOP error code	1
Hardware status code	2
Application status code	3

• STOP error code has the highest priority.
• The priority of a hardware status code differs depending on its cause. For details, see "4.4.2   Status codes to be displayed".

 (2) In the application status display mode.

Code class	Priority order
STOP error code	1
Application status code	2

• STOP error code has the highest priority.

4.4.5   Status display digital LEDs control functions

The following functions are offered as library functions for controlling the status display digital LEDs. For details about the library functions, see "6.1   RAS Library Interface".

● Display an application status code: Use the SetStCode7seg function.
● Turns off the currently displayed application status code: Use the TurnOff7seg function.
● Sets the status display mode: Use the SetMode7seg function.

4.5 Remote Notification

4.5.1 Overview

If you use this function, from a remote device through the network, you can check hardware conditions that can only be checked beside this equipment without this function. Even when hardware conditions cannot be checked beside this equipment because, for example, the system administrator is away from this equipment or this equipment is built into the facility, the hardware conditions can be checked from a remote device.

This function uses Simple Network Management Protocol (SNMP) to notify hardware conditions. This allows you to use commercially available network management software that supports SNMP and to monitor distributed instances of this equipment and other devices all from one location.

NOTE

- The remote notification uses SNMP, a protocol in the application layer of the TCP/IP, and User Datagram Protocol (UDP) in the transport layer. That means if the network load is high, hardware conditions may not be received properly.
- The remote notification uses the SNMP service, a standard feature of Windows®. For information about how to enable the standard Windows® SNMP service, see "4.5.3 Enabling the remote notification".

4.5.2   Hardware conditions that can be acquired by using remote notification

The following hardware conditions and settings can be acquired from a remote device:

• Fan condition

• Temperature condition inside the chassis

• Drive condition

• Memory condition

• RAID status (B Model only)

• RAS function settings

• Operational mode (normal mode)

• Version information of the extended Management Information Base (MIB) for HF-W.

The following transitions of the hardware conditions are notified by using a trap.

(1) Fan condition

     • Normal → Error

     • Error → Normal

(2) Temperature condition inside the chassis

     • Normal → Error

     • Error → Normal

(3) Drive condition

     • Normal → Failure anticipated

     • Normal → Used hours exceeded the threshold

(4) Memory condition

     • Normal → Frequent error correction

     • Frequent error correction → Normal

(5) RAID status (B Model only)

     • Optimal → Error (Degrade, Unknown)

     • Error (Degrade, Unknown) → Optimal

(6) Operational mode

     • HF-W stopped → Started in normal mode

     • Running in normal mode → Running in simulation mode
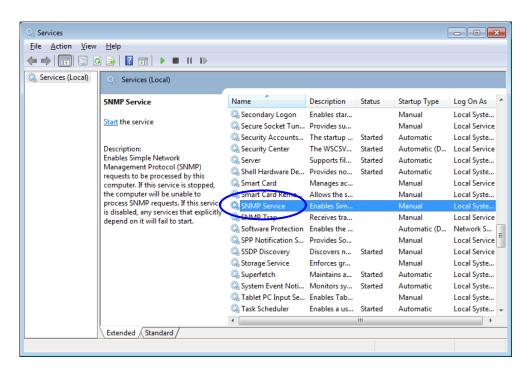
4.5.3   Enabling the remote notification

This function is disabled in the default factory-shipped settings. The remote notification uses the standard Windows® SNMP service. If you enable the SNMP service, the remote notification is enabled.
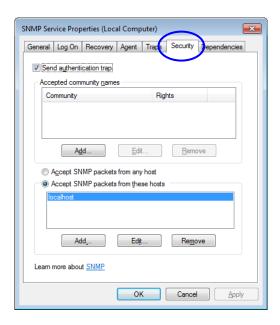
When you use the remote notification, follow the procedure below to enable the SNMP service:

(1) Starting the "**SNMP Service Properties**" window
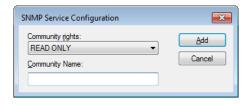
1. If you are not logged on to the computer as an administrator, log on to the computer as an administrator.
2. Follow the procedure below to start the **Services** window.
   • Open **Control Panel**.
   • Double-click **System and Security** > **Administrative Tools** > **Services**.

3. Double-click **SNMP Service** to start the "**SNMP Service Properties**" window.

(2) SNMP security configuration

1. In the "**SNMP Service Properties**" window, select the **Security** tab.

2. If you want to send a trap message whenever authentication fails, select the "**Send authentication trap**" check box.

3. Under "**Accepted community names**," click **Add**. The "**SNMP Service Configuration**" window is displayed. In the "**Community rights**" list, select **READ ONLY**. In the **Community Name** box, type the community name you want, and click **Add**.
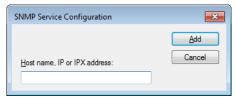
4. Specify whether to accept SNMP packets from hosts.

If you want to accept SNMP packets from any manager on the network:

• Select **Accept SNMP packets from any host**.
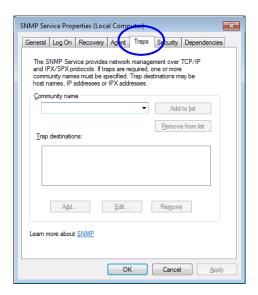
If you want to restrict SNMP packets:

• Select **Accept SNMP packets from these hosts**.

• Click **Add**.

• The "**SNMP Service Configuration**" window is displayed. Enter the host name, IP or IPX address of a host you want to accept the SNMP packets from, and click **Add**.

5. In the "**SNMP Service Properties**" window, click **Apply**.

(3) SNMP trap configuration

    1. In the "**SNMP Service Properties**" window, select the **Traps** tab.
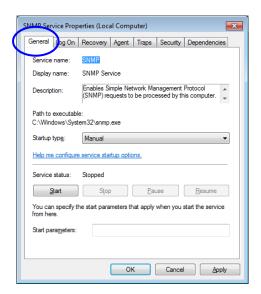
    2. Under **Community name**, type the name of the community that trap messages are sent to, and click "**Add to list**."

    3. Under **Trap destinations**, click **Add**. The "**SNMP Service Configuration**" window is displayed. Enter the host name, IP or IPX address of a destination you want to send traps to, and click **Add**.

    4. In the "**SNMP Service Properties**" window, click **Apply**.

(4) Starting the SNMP service

   1. In the "**SNMP Services Properties**" window, select the **General** tab.
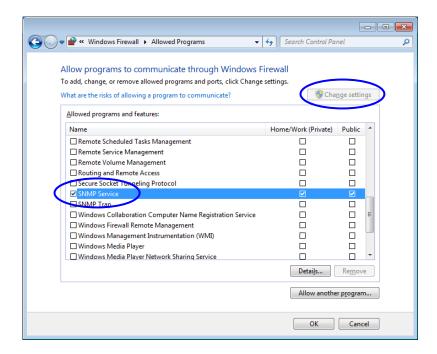
   2. Click **Start**. The SNMP service starts and the remote notification for the hardware status is enabled.

   3. In order to start the SNMP service automatically at the next OS startup, in the **Startup type** list, select **Automatic**.

   4. In the "**SNMP Service Properties**" window, click **OK**.

—— NOTE ——

• If when the SNMP starts, there is an error that is configured to be notified by using a trap, the trap is sent at the timing when the SNMP starts.

• If Windows Firewall is configured to block the SNMP service, you cannot acquire the hardware status from a remote device. If you set up the firewall to block SNMP service, follow the procedure below to undo it.
But, SNMP service can pass through Windows Firewall by default, you do not have to go through the following procedure.

1. If you are not logged on to the computer as an administrator, log on to the computer as an administrator.
2. Open **Control Panel**, click "**System and Security**."
3. Under **Windows Firewall**, click "**Allow a program through Windows Firewall**."

4. The "**Allowed Programs**" window appears. Click **Change settings**, and under "**Allowed programs and features**," select the "**SNMP Service**" check box.

5. Click **OK**.

4.5.4   Objects in the extended MIB for HF-W

In order to acquire the hardware status of this equipment from a remote device, use the extended MIB for HF-W. This subsection provides a list of objects defined in the extended MIB for HF-W and a description of those objects.

(1) Objects related to the hardware conditions and settings

Table 4-11 shows a list of the objects related to the hardware status and a description of those objects. The object ID of each object is obtained by either replacing the following "x" with the object in the following table or replacing the following "y" with the object number in the table.

Object ID: **.iso.org.dod.internet.private.enterprises.Hitachi.systemExMib.**

                hfwExMib.hfwRasStatus.x (x is an object in the table below)

                or

                .1.3.6.1.4.1.116.5.45.1.y (y is an object number in the table below)

Table 4-11   Objects Related to the Hardware Status

(1/2)

No.	Object	Object number	Description	Description of the values
1	hfwFan	1	Fan condition group	-
2	hfwFan.fanNumber	1.1	The number of monitored fans	-
3	hfwFan.fanTable.fanEntry.fanIndex	1.2.1.1	Index number of fanEntry	-
4	hfwFan.fanTable.fanEntry.fanName	1.2.1.2	Fan name	PS fan: Power fan System fan: System fan CPU fan: CPU fan
5	hfwFan.fanTable.fanEntry.fanStatus	1.2.1.3	Fan condition	1: Normal 2: Error
6	hfwTemp	2	Temperature condition group	-
7	hfwTemp.tempNumber	2.1	Number of monitored temperatures	-
8	hfwTemp.tempTable.TempEntry.tempIndex	2.2.1.1	Index number of tempEntry	-
9	hfwTemp.tempTable.TempEntry.tempName	2.2.1.2	Name of the temperature to bemonitored	Internal temperature: Temperature inside the chassis
10	hfwTemp.tempTable.TempEntry.tempStatus	2.2.1.3	Temperature condition	1: Normal 2: Error
11	hfwHdd	3	Drive condition group	-
12	hfwHdd.hddNumber	3.1	Number of monitored drive bays	-
13	hfwHdd.hddTable.hddEntry.hddIndex	3.2.1.1	Index number of hddEntry	-

(2/2)

No.	Object	Object number	Description	Description of the values
14	hfwHdd.hddTable.hddEntry.hddStatus	3.2.1.2	Drive condition	1: Healthy 2: Not mounted 3: Failure anticipated 5: Used hours exceeded the threshold 7: Offline 8: Rebuild 12:Incomplete Data 99: Unknown
15	hfwHdd.hddTable.hddEntry.hddUseTime	3.2.1.3	Drive power-on (=used) hours (in hours)	-
16	hfwMem	5	Memory condition group	-
17	hfwMem.memNumber	5.1	Number of monitored memory slots	-
18	hfwMem.memTable.memEntry.memIndex	5.2.1.1	Index of memEntry	-
19	hfwMem.memTable.memEntry.memName	5.2.1.2	Memory name	DIMM A1: DIMM A1 slot DIMM B1: DIMM B1 slot
20	hfwMem.memTable.memEntry.memStatus	5.2.1.3	Memory condition	1: Normal 2: Error (Frequent error correction) 3: Not mounted
21	hfwRaid	4	RAID status group	-
22	hfwRaid.raidNumber	4.1	Number of monitored arrays	-
23	hfwRaid.raidTable.raidEntry.raidIndex	4.2.1.1	Index number of raidEntry	-
24	hfwRaid.raidTable.raidEntry.raidStatus	4.2.1.2	RAID status	1: Optimal 2: Degrade 4: Unknown
25	hfwRaid.raidTable.raidEntry.raidLevel	4.2.1.3	RAID level	1：RAID1
26	hfwRaid.raidTable.raidEntry.raidProgress	4.2.1.4	Progress of the rebuild process (in %)	-
27	hfwRaid.raidTable.raidEntry.raidMediaError	4.2.1.5	Whether media error occurred	1: No media error occurred. 2: Media error occurred.

No.2: For this equipment, the value is set to 3.
No.7: For this equipment, the value is set to 1.
No.12: The number of internal drives that can be mounted in the HF-W is set for the number of monitored drive bays.
　　　For this equipment, the number is 2.
No.17: The number of memory slots is set for the number of monitored memory slots.
　　　For this equipment, the number is 2.
No.22: Other than B Model, the value is set to 0.
Nos. 23 to 27: Supported by B Model only.

Table 4-12 shows a list of the objects related to the RAS function settings and a description of those objects. The object ID of each object is obtained by either replacing the following "x" with the object in the following table or replacing the following "y" with the object number in the table.

Object ID: .iso.org.dod.internet.private.enterprises.Hitachi.systemExMib.

hfwExMib.hfwRasSetting.x (x is an object in the table below)

or

.1.3.6.1.4.1.116.5.45.2.y (y is an object number in the table below)

Table 4-12   Objects Related to the RAS Function Settings

No.	Object	Object number	Description	Description of the values
1	hfwFanAutoShutdown	1	Automatic shutdown when a fan failure occurred	1: Enabled 2: Disabled
2	hfwTempAutoShutdown	2	Automatic shutdown when the temperature is abnormal	1: Enabled 2: Disabled
3	hfwRemoteShutdown	3	Automatic shutdown when a remote shutdown is requested	1: Enabled 2: Disabled

Table 4-13 shows a list of the objects related to operational modes and a description of those objects. The object ID of each object is obtained by either replacing the following "x" with the object in the following table or replacing the following "y" with the object number in the table.

Object ID: .iso.org.dod.internet.private.enterprises.Hitachi.systemExMib.

hfwExMib.hfwRasInfo.x (x is an object in the table below)

or

.1.3.6.1.4.1.116.5.45.3.y (y is an object number in the table below)

Table 4-13   Objects Related to Operational Modes

No.	Object	Object number	Description	Description of the values
1	hfwRasMode	1	Operation mode	1: Normal mode 2: Simulation mode

Table 4-14 shows a list of the objects related to the version information of the extended MIB for HF-W and a description of those objects. The object ID of each object is obtained by either replacing the following "x" with the object in the following table or replacing the following "y" with the object number in the table.

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

Table 4-14   Objects Related to the Extended MIB for HF-W

No.	Object	Object number	Description	Description of the values
1	Version	1	Version number of the extended MIB for HF-W	-
2	Revision	4	Revision number of the extended MIB for HF-W	-

(2) Objects related to the trap notification

Table 4-15 shows a list of the objects related to the trap notification when an error occurs as well as a description and notification data for those objects. The enterprise ID for the trap notification when an error occurs 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 4-15   Objects Related to the Trap Notification (When an Error Occurs)

No.	Object	Trap number	Description	Notification data	
				Object used	Value
1	hfwFanError	1	A fan failure occurs.	fanName	Name of the failed fan
				fanStatus	2: Error
				hfwFanStMsg	Revolution of %1 deteriorated remarkably.
2	hfwTempError	2	The temperature inside the chassis becomes abnormal.	tempName	Internal temperature
				tempStatus	2: Error
				hfwTempStMsg	Internal temperature exceeded prescribed value.
3	hfwSmartDetect	3	A failure is anticipated for the drive.	hddIndex	The drive bay number of the drive for which a failure is anticipated by SMART
				hddStatus	3: Failure anticipated
				hfwSmartStMsg	A failure may be imminent on the drive of the drive bay%2.
4	hfwHddOverRun	4	The drive power-on (=used) hours exceeded the threshold.	hddIndex	The drive bay number of the drive for which the power-on hours exceeded the threshold
				hddStatus	5: Used hours exceeded the threshold.
				hfwHddUseTimeStMsg	Used hours on the drive of the drive bay%2 exceeded prescribed value.
5	hfwMemError	6	A frequent error correction occurs.	memName	Name of the memory slot with frequent error correction
				memStatus	2: Error (Frequent error correction)
				hfwMemStMsg	In the %3, error correcting have occurred with high frequency.
6	hfwArrayError	7	A RAID error occurs.	raidIndex	1
				hfwArrayStMsg	An error occurred on Array%4.

No.1: %1 denotes the name of the failed fan.
Nos. 3 and 4: %2 denotes the drive bay number.
No.5: %3 denotes the name of the memory slot with frequent error correction.
No.6: %4 denotes the number of the array with an error. For this equipment, the array number is always 1.

Table 4-16 shows a list of the objects related to the trap notification when the equipment has recovered from an error and a description of those objects. The enterprise ID for the trap notification when the equipment has recovered from an 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 4-16   Objects Related to the Trap Notification (When the Equipment Has Recovered from an Error)

No.	Object	Trap number	Description	Notification data	
				Object used	Value
1	hfwFanRecover	1	Recovery from fan failure	fanName	Name of the recovered fan
				fanStatus	1: Normal
				hfwFanStMsg	Revolutions of %1 returned to normal value.
2	hfwTempRecover	2	Recovery from abnormal temperature in the chassis	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 recovered memory slot
				memStatus	1: Normal
				hfwMemStMsg	In the %2, frequency of the error correctings deteriorated.
4	hfwArrayRecover	7	Recovery from RAID error	raidIndex	1
				hfwArrayStMsg	Array%3 is restored.

No.1: %1 denotes the name of the recovered fan.
No.3: %2 denotes the name of the memory slot that recovered from frequent error correction.
No.4: %3 denotes the number of the array that recovered from an error. For this equipment, the array number is always 1.

Table 4-17 shows a list of the objects related to the trap notification when the equipment has started in normal mode and a description of those objects. The enterprise ID for the trap notification related to operational 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 4-17   Objects Related to the Trap Notification (Operational Modes)

No.	Object	Trap number	Description	Notification data	
				Object used	Value
1	hfwRasService Started	1	Startup in normal mode	hfwRasMode	1: Normal mode
				hfwRasStartMsg	RAS Service is running.
2	hfwSimulation ModeStarted	2	Transition to simulation mode	hfwRasMode	2: Simulation mode
				hfwRasStartMsg	RAS Service switched to Simulation Mode.

4.5.5   Extended MIB file for HF-W
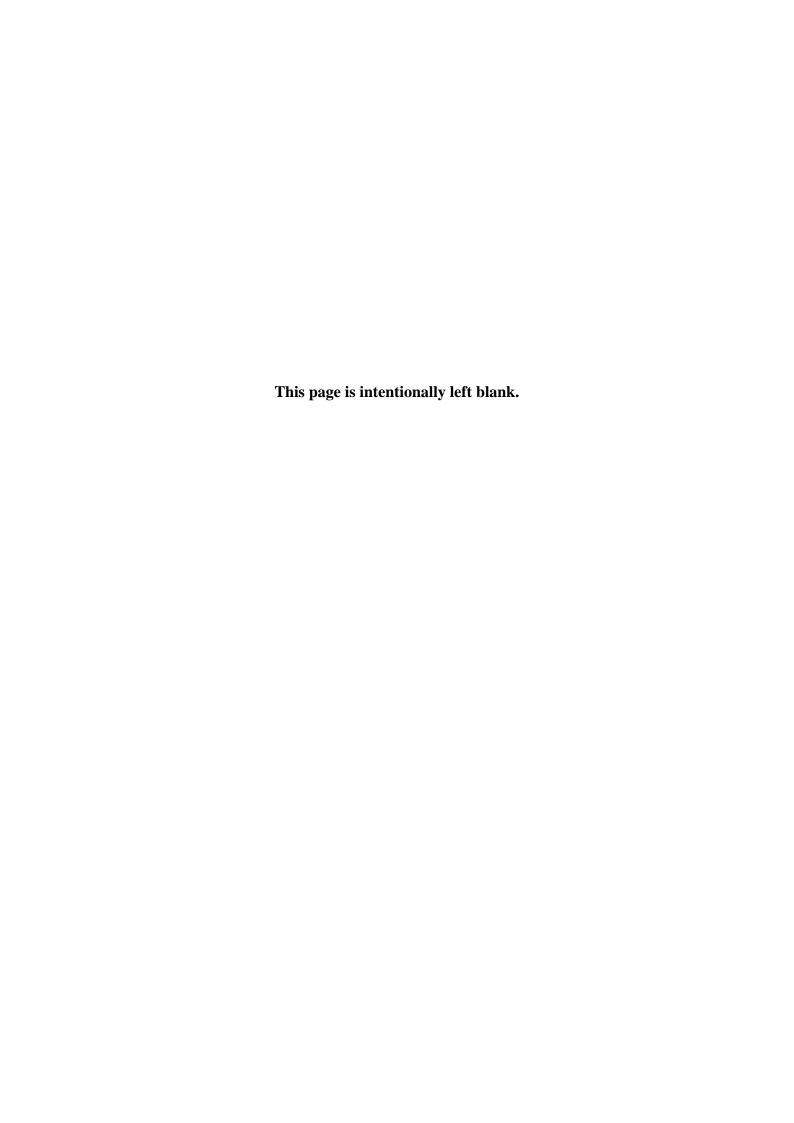
The extended MIB file for HF-W is as follows.

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

4.6   Status Acquisition by Using the RAS Library

By using the RAS library functions, the following hardware conditions can be acquired. For details about the library functions, see "6.1   RAS Library Interface".

● To acquire the memory condition: Use the GetMemStatus function.

● To acquire the failure prediction status of the drive: Use the GetHddPredict function.

● To acquire the drive condition: Use the hfwDiskStat function.

● To acquire the RAID status: Use the hfwRaidStat and RaidStat functions. [B Model only]

This page is intentionally left blank.

CHAPTER 5   CONTROLLING THE HARDWARE

The RAS feature can have the following controls over this equipment.

(1) Automatic shutdown of the equipment
    When a hardware error occurs or a remote shutdown request through the contact input is
    detected, the equipment can be automatically shut down. For details, see"5.1   Automatic
    Shutdown of the Equipment".

(2) Controlling the hardware by using the RAS library
    A user application can control the hardware of this equipment by using the RAS library. For
    details, see "5.2   Controlling the Hardware by Using the RAS Library".

(3) Controlling the RAID status using the RAID configuration control command
    The RAID configuration control command displays the status of the RAID and its drives. The
    command can forcibly disconnect a drive and change the setting for RAID. For details, see
    "5.3   RAID Configuration Control Command (raidctrl) [B Model only]".

5.1   Automatic Shutdown of the Equipment

This function automatically shuts down the equipment when running the equipment would pose a danger when fan failure or abnormally high temperature is detected. By doing so, you can protect internal parts such as a processor from thermal degradation and prevent thermal runaway of the system due to malfunction of this equipment. The equipment can be also automatically shut down by a remote shutdown signal input from an external device.

5.1.1   Automatic shutdown when detecting fan failure

When the equipment detects an error in one of the fans, the equipment automatically shuts down.

• This function can be enabled or disabled in the **RAS Setup** window. In the default factory-shipped setting, this function is enabled. For details, see "3.1.3   Using the RAS Setup window".

• After an automatic shutdown initiated by this function is complete, the power is turned off.

• Alternatively; a user application can detect a fan failure using a RAS event and shut down the equipment. For information about a RAS event, see "4.2   RAS Event Notification".

NOTICE

● If this equipment continues to operate with fan failure detected, internal parts such as a processor are not cooled down sufficiently, and that may cause thermal runaway of the system due to malfunction of the equipment or result in damage to the parts. If possible, enable the automatic shutdown feature.

● If the automatic shutdown feature is not used, have a user application detect a fan failure using a RAS event and shut down the equipment.

—— NOTE ——

For information about how to replace a fan, refer to "HF-W2000 Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".

5.1.2   Automatic shutdown when detecting abnormally high temperature
When the temperature sensor in this equipment detects the temperature is abnormally high inside the chassis, the equipment automatically shuts down.

• This function can be enabled or disabled in the **RAS Setup** window. In the default factory setting, this function is disabled. For details, see "3.1.3   Using the RAS Setup window".

• After an automatic shutdown initiated by this function is complete, the power is turned off.

• Alternatively; a user application can detect abnormally high temperature using a RAS event and shut down the equipment. For information about a RAS event, see "4.2   RAS Event Notification".

──── NOTE ────

• If the temperature inside the chassis is high, parts can be degraded rapidly due to heat. Then it is not advisable to continue to use this equipment also in terms of the life span of the equipment. On the other hand, if the temperature is abnormally high when the fans are working properly, the abnormally high temperature must be caused by an external factor such as malfunctioning air conditioning at the location of the equipment. You can isolate the cause of the abnormally high temperature while the equipment is running. Because of this, this function is disabled in the default factory-shipped setting,

• If you continue to operate this equipment after abnormally high temperature is detected and the temperature further rises to a dangerous level, the equipment is forcibly shut down and the power is turned off regardless of whether this function is disabled.

5.1.3   Automatic shutdown when a remote shutdown request through the contact input is detected
If this function is enabled, this equipment is automatically shut down when the remote shutdown contact (RMTSTDN contact) in the RAS external contacts interface is closed. By using this function, you can shut down this equipment from a remote location.

• This function can be enabled or disabled in the **RAS Setup** window. In the default factory-shipped setting, this function is enabled. For details, see "3.1.3   Using the RAS Setup window".

• After an automatic shutdown initiated by this function is complete, the power is turned off.

• Alternatively; a user application can detect a remote shutdown request through the contact input using a RAS event and shut down the equipment. For information about a RAS event, see "4.2   RAS Event Notification".

5.2 Controlling the Hardware by Using the RAS Library

By using the RAS library functions, you can shut down the system and control the general purpose external contacts and the status display digital LEDs. For details about the library functions, see "6.1 RAS Library Interface".

- To shut down the system: Use the BSSysShut function.
- To control the watchdog timer: Use the WdtControl function.
- To control the general purpose external contact outputs: Use the GendoControlEx and GendoControl functions.
- To control the general purpose external contact inputs: Use the GetGendiEx and GetGendi functions.
- To control the status display digital LEDs: Use the SetStCode7seg, TurnOff7seg, and SetMode7seg functions.

5.3   RAID Configuration Control Command (raidctrl) [B Model only]

The raidctrl command displays the status of the RAID and its drives. The command can also forcibly disconnect a drive and change the setting for RAID. This command can be launched at the command prompt. The following shows how to use this command.

<Name>
    raidctrl - RAID drive status display and configuration control, setting update

<Syntax>
    raidctrl   [/OFFLINE DRVNO] [/NOTIFY [{ON|OFF}]] [/INCOMPLETE [{ON|OFF}]]
                [/REBUILD] [/LOAD [{H|M|L}]] [/MANUAL [{ON|OFF}]]

<Description of the function>
    The raidctrl command displays the status of the drives in the RAID, controls the configuration of those drives, and changes the setting for RAID. This command offers one of the following functions depending on the specified options.

    • With no options:
            The status of the RAID and the drives in the RAID will be displayed.
    • When /OFFLINE option is specified:
            The specified drive is forcibly disconnected and set to offline.
    • When /NOTIFY option is specified:
            You can switch whether notification is sent through the interface of this equipment when a media error occurs.
    • When /INCOMPLETE option is specified:
            You can select whether to disconnect a drive if there is any possibility of data mismatch between the drives.
    • When /REBUILD option is specified:
            A rebuild process is started in all drives ready for the process.
    • When /LOAD option is specified:
            You can specify the level of the load on the system during a rebuild process.
    • When /MANUAL option is specified:
            You can select whether to start a rebuild process manually or automatically when a drive is replaced.

NOTE

- You need to have administrator privileges to execute this command except for no options.
  Start the command prompt as an administrator, and then execute the raidctrl command.
  In addition, you cannot execute those actions from a remote session except for no options.
- Do not run multiple instances of this command at the same time.

---

(1) Displaying the status of the RAID and its drives (with no options)

If you execute the raidctrl command without any options, the command will output the status of the RAID and the drives in the RAID on this equipment. Tables 5-1 and 5-2 show the status of the RAID and drives displayed in the output.

Table 5-1   RAID Status Displayed in the Output of the raidctrl Command

No.	Displayed Status	Description
1	OPTIMAL	The RAID is working properly.
2	DEGRADE	The RAID is being degraded.
3	DEGRADE (REBUILD xx%)	The RAID is under a rebuild process.
4	UNKNOWN	Unknown status

Nos.1 through 3: If a media error occurs, "MEDIA ERROR" is attached to the end of the displayed status.
(Example) If a media error occurs when the RAID is operating normally:
OPTIMAL (MEDIA ERROR)

Table 5-2   Drive Status Displayed in the Output of the raidctrl Command

No.	Displayed Status	Description
1	ONLINE	The drive is working properly.
2	OFFLINE	The drive is disconnected from the RAID.
3	INCOMPLETE DATA	The drive is disconnected from RAID because of data mismatch between drives.
4	REBUILD	The RAID is under a rebuild (copy) process.
5	NOT CONNECTED	The drive is not connected.
6	UNKNOWN	Unknown status

The following is an example of the output when the raidctrl command is executed without options.

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

(2) Disconnecting a drive (with /OFFLINE option)

If you execute the raidctrl command with the /OFFLINE option, the specified drive is forcibly disconnected and set to offline.

This option can be used only when the RAID status is "OPTIMAL". Only 1 or 2 can be specified for DRVNO. If you specify other values, an error message is displayed and the command exits.

The following shows an example of executing the raidctrl command to disconnect a drive bay. When a message is displayed to confirm whether to disconnect, type **y** to actually disconnect the drive bay. If you type **n**, disconnecting the drive bay is canceled, and the command exits.

```
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 ──────────────────────────────────

A disconnected drive cannot be set to online and used again.
Make sure that you select the correct drive.

(3) Switching whether a media error is notified when it occurs (with the /NOTIFY option)

If you execute the raidctrl command with the /NOTIFY option, you can switch whether media errors are notified when they occur.

If the /NOTIFY option is specified alone, the current notification setting is displayed. In the default factory settings, a media error is not notified.

The following are examples of executing the command with the /NOTIFY option.
• To check the media error notification setting:
  raidctrl /NOTIFY

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

• To enable media error notification:
  raidctrl /NOTIFY ON
• To disable media error notification:
  raidctrl /NOTIFY OFF

(4) Switching when incomplete data occurs (with the /INCOMPLETE option)

If you execute the raidctrl command with the /INCOMPLETE option, you can select whether to disconnect or continue using the drive in the drive bay2 if there is any possibility of data mismatch between the drives due to, for example, unexpected power shutdown. If the /INCOMPLETE option is specified without "ON" or "OFF", the current setting is displayed. The default factory setting is "OFF" (continue using the drive).

This setting is enabled after the restart.

The following are examples of executing the command with the /INCOMPLETE option.

• To check the incomplete data setting:

  raidctrl /INCOMPLETE

```
C:\ >raidctrl /INCOMPLETE
Incomplete Data Setting:OFF
C:\ >
```

• To disconnect the drive if there is any possibility of data mismatch:

  raidctrl /INCOMPLETE ON

• To continue using the drive even if there is any possibility of data mismatch:

  raidctrl /INCOMPLETE OFF

(5) Starting a rebuild process (with the /REBUILD option)

If you execute the raidctrl command with the /REBUILD option, the rebuild process starts when it is possible to do so. If the /REBUILD option is specified, the rebuild process starts under one of the following conditions.

- The drive in drive bay2 is disconnected because the possibility of data mismatch between drives is detected. (That is, the drive status of the drive bay1 is "ONLINE", and the drive status of drive bay2 is "INCOMPLETE DATA".)
- A new drive is installed when the rebuild start method setting is "MANUAL". (That is, one of the drive statuses of the two drives is "ONLINE", and the other is "OFFLINE".)

── NOTE ──────────────────

- Even when one of the drive statuses is "ONLINE" and the other is "OFFLINE", the rebuild process does not start for the "OFFLINE" drive if the drive status of the drive was set to "OFFLINE" due to a failure. A rebuild process starts for an "OFFLINE" drive only if the drive status of the drive was set to "OFFLINE" because the drive was newly installed.

The following are examples of executing the command with the /REBUILD option.

```
C:\ >raidctrl /REBUILD
This might take several seconds to start rebuild.
Are you sure you want to continue with this process? [y/n] y

This command completeed successfully.
C:\ >
```

── NOTE ──────────────────

- The rebuild process takes time to start. If the rebuild process does not start within 3 minutes, check the system log in the event log of Windows®.

(6) Setting the load during a rebuild process (with the /LOAD option)

If you execute the raidctrl command with the /LOAD option, you can set the level of the write load on the system to the drive during a rebuild process. If the /LOAD option is specified without "H", "M", or "L", the current setting is displayed. You can adjust the write load by changing this setting. However, be aware that, the lower the load, the longer the rebuild process. When you change this setting, the new setting is effective after this equipment restarts. The default factory setting is "H" (perform a rebuild process with a high load).

The following are examples of executing the command with the /LOAD option.

• To check the load setting under rebuild:

raidctrl /LOAD

```
C:\ >raidctrl /LOAD
Rebuild Load Level:HIGH
C:\ >
```

• To select a high load (HIGH):

raidctrl /LOAD H

• To select a medium load (MIDDLE):

raidctrl /LOAD M

• To select a low load (LOW):

raidctrl /LOAD L

(7) Switching the rebuild start method (with the /MANUAL option)

If you execute the raidctrl command with the /MANUAL option, you can select whether to start a rebuild process manually when a drive is replaced. If the /MANUAL option is specified without "ON" or "OFF", the current setting is displayed. The default factory setting is "OFF" (a rebuild process is started automatically).

This setting is enabled after the restart.

The following are examples of executing the command with the /MANUAL option.

• To check the setting of the way to start rebuild:

raidctrl /MANUAL

```
C:\ >raidctrl /MANUAL
Manual Rebuild:OFF
C:\ >
```

• To start a rebuild process manually after a drive is replaced:

raidctrl /MANUAL ON

• To start a rebuild process automatically after a drive is replaced:

raidctrl /MANUAL OFF

<Diagnosis>

When the raidctrl command finishes normally, it returns exit code 0.

If the command returns abnormally, one of the error messages listed in Table 5-3 is

displayed, and the command returns an exit code other than 0.

Table 5-3   Error Messages of the raidctrl Command

No.	Error message	Description
1	Usage:raidctrl [/OFFLINE DRVNO] [/NOTIFY [{ON|OFF}]] [/INCOMPLETE [{ON|OFF}]] [/REBUILD] [/LOAD [{H|M|L}]] [/MANUAL [{ON|OFF}]]	There is an error in the specified options. Specify correct options.
2	Specified Drive is invalid.	The drive is not a valid drive status.
3	No valid drive found.	There was no valid drive.
4	An error occurred in %1. errorcode = %2	An unexpected error (%2) occurred in the function (%1). Retry the command. If the same error message persists, restart this equipment.
5	You do not have the privilege to execute this command option. Please execute this command option again on "Administrator: Command Prompt".	You do not have administrator privileges. Log on to the computer using an administrator account and run the command again. If User Account Control (UAC) is enabled, start the command prompt with administrator privileges and run the command.
6	Access denied. Log on console session, And execute this command option again on "Administrator: Command Prompt".	You cannot use this command with this option from a remote session. Log on to a console session and run the command again. If User Account Control (UAC) is enabled, start the command prompt with administrator privileges and run the command.
7	raidctrl: This function was executed on a non-RAID model.	The command is executed on a model other than B Model. This command can be executed on B Model only.

CHAPTER 6   LIBRARY FUNCTIONS

A user application can get and control the hardware status of this equipment by using the RAS library.

For information about the hardware specifications of the RAS external contacts interface described in this chapter and the usage of each contact, refer to "HF-W2000 Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".

6.1   RAS Library Interface

6.1.1   Overview

This chapter describes the interface to the functions provided by the RAS library.

Table 6-1 shows a list of RAS library functions.

Table 6-1   RAS Library Functions

No.	Function name	Use	DLL
1	BSSysShut	Shuts down the equipment.	w2kras.dll
2	WdtControl	Retriggers, gets the status of, and stops the watchdog timer.	
3	GendoControl	Opens and closes the general purpose external contact output (GENDO0). (for backward compatibility)	
4	GendoControlEx	Opens and closes the general purpose external contact outputs (GENDO0, GENDO1, and GENDO2).	
5	GetGendi	Gets the status of the general purpose external contact input (GENDI). (for backward compatibility)	
6	GetGendiEx	Gets the status of the general purpose external contact inputs (GENDI, GENDI0, GENDI1, and GENDI2).	
7	MconWriteMessage	Records a specified message (characters) in this equipment's own log files.	
8	GetMemStatus	Acquires the condition of the memory built into this equipment.	
9	GetHddPredict	Acquires the failure prediction status of the drive. (for backward compatibility)	
10	hfwDiskStat	Acquires the status of the drive.	hfwras.dll
11	hfwRaidStat	Acquires the RAID status. [B Model Only]	
12	RaidStat	Acquires the status of the drives in the RAID. (for backward compatibility) [B Model Only]	hfwraid.dll
13	SetStCode7seg	Outputs an application status code on the status display digital LEDs.	ctrl7seg.dll
14	TurnOff7seg	Clears an application status code on the status display digital LEDs.	
15	SetMode7seg	Configures the status display mode of the status display digital LEDs.	

Those functions mentioned above are offered in the DLLs (w2kras.dll, hfwras.dll, hfwraid.dll, and ctrl7seg.dll).

─── NOTE ───

・If the operating system is a 64-bit version, those DLLs can be linked into only 64-bit applications. 32-bit applications cannot use those DLLs.
The following functions have been provided to maintain backward compatibility with Windows NT® programs and old equipment, but 64-bit versions do not support these functions. Use the function provided as an alternative.

Not used function by 64-bit operating systems	Alternative function
GetHddPredict	hfwDiskStat
RaidStat	hfwRaidStat

・Do not copy or move w2kras.dll, hfwras.dll, hfwraid.dll, and ctrl7seg.dll to another directory. If you do, the RAS feature of this equipment cannot run properly.

The functions offered by w2kras.dll and ctrl7seg.dll can be called from Visual Basic® (.NET is required for 64-bit operating systems). When you call functions number 1 through 7, 9, and 13 through 15 from Visual Basic®, add "_VB" to the end of the name of each function. Function parameters are the same. For example, when you call the WdtControl function from Visual Basic®, use the function name "WdtControl_VB".

The following files are provided as import libraries:
　　%ProgramFiles%\HFWRAS\lib\w2kras.lib
　　%ProgramFiles%\HFWRAS\lib\hfwras.lib
　　%ProgramFiles%\HFWRAS\lib\ctrl7seg.lib
　　%ProgramFiles%\HFWRAS\lib\hfwraid.lib
When you use a library, link the corresponding import library.

The following files are provided as header files for the libraries:
　　%ProgramFiles%\HFWRAS\include\w2kras.h
　　%ProgramFiles%\HFWRAS\include\hfwras.h
　　%ProgramFiles%\HFWRAS\include\ctrl7seg.h
　　%ProgramFiles%\HFWRAS\include\hfwraid.h
When you use a library in C, include the corresponding header file.

<Migration from a Windows® XP application>
　　・If the operating system is a 32-bit version, you can execute the application as is.
　　・If the operating system is a 64-bit version, you must rebuild the application as a 64-bit application.

6.1.2　Shutdown function (BSSysShut)

<Name>

BSSysShut - System shutdown

<Syntax>

#include <w2kras.h>

int BSSysShut(reboot)

int reboot:　　/*Reboot flag*/

<Description>

BSSysShut shuts down the system.

The reboot argument is used for specifying whether to reboot the system after the shutdown.

reboot = 0: The power to this equipment is turned off after the shutdown.

reboot ≠ 0: The system reboots after the shutdown.

<Diagnosis>

0: Successful completion (System shutdown process has started)

1: Shutdown privilege acquisition error

2: Internal error (OS shutdown failed)

<Sample program>

We provide a sample program that uses this function in C. For information about the name of the sample program and where you can find it, see "6.2　Sample Programs".

6.1.3   Watchdog timer control function (WdtControl)

(1) Function interface

<Name>

WdtControl - Watchdog timer control / status acquisition

<Syntax>

#include <w2kras.h>
BOOL WdtControl(DWORD dwCmd, PDWORD pdwCount);

<Description>

This function performs the action specified by dwCmd on the watchdog timer.
In order to use this function, in the **RAS Setup** window, select **Retriggered by application program** under **Watchdog timer setting**. If the watchdog timer setting is different, this function terminates with an error. If you call the GetLastError Windows API function, an error code W2KRAS_WDT_NONMANUAL is returned.
Each parameter of this function is explained below.

dwCmd:

This parameter specifies which action is performed on the watchdog timer. The following options are available for this parameter.

Table 6-2   List of Actions of WdtControl Specified by dwCmd

dwCmd	Explanation of the action
WDT_SET (0x00)	Specifies the timeout (in seconds).
WDT_STOP (0x01)	Stops the watchdog timer.
WDT_STAT (0x02)	Acquires the status of the watchdog timer.

If a value other than the above is specified, this function terminates with an error. If you call the GetLastError Windows API function, an error code W2KRAS_INVALID_PARAMETER is returned.

pdwCount:

If dwCmd is WDT_SET, you can configure the timeout of the watchdog timer by specifying the timeout of the watchdog timer in the variable pointed by pdwCount and calling this function.
Specify a value between 1 and 63 in seconds. If a value other than the above is specified, this function terminates with an error. If you call the GetLastError Windows API function, an error code W2KRAS_INVALID_PARAMETER is returned.
When this function returns, the value of the variable pointed by pdwCount is undefined. Do not use the value.
If dwCmd is WDT_STOP, the value of pdwCount is ignored. When this function returns, the value of the variable pointed by pdwCount is undefined. Do not use the value.

If dwCmd is WDT_STAT, the remaining time (in seconds) at the timing of the function call until the watchdog timer expires is stored to the variable pointed by pdwCount. If the value of the variable pointed by pdwCount is 0 when the function returns, that means the watchdog timer has already timed out.

Note that the value of the variable pointed by pdwCount when the function is called is ignored.

<Diagnosis>

If this function completes successfully, the function returns TRUE. If this function terminates with an error, the function returns FALSE.

When this function terminates with an error, call the GetLastError Windows API function to get the error code. Error codes returned by this function on its own are as follows.

Error code (value)	Description
W2KRAS_INVALID_PARAMETER (0x2001)	There is an error in the specified parameters.
W2KRAS_WDT_NONMANUAL (0x2002)	This function cannot be used because **Retriggered by application program** is not selected under **Watchdog timer setting** in the **RAS Setup** window.
W2KRAS_NOT_INITIALIZE (0x2005)	Startup of the RAS software has not completed yet.
W2KRAS_INTERNAL_ERROR (0x2007)	An internal error has been generated.

Other error codes come from the Windows API functions used by this function. For details about those error codes, refer to the Windows API help.

<Sample program>

We provide a sample program that uses this function in C. For information about the name of the sample program and where you can find it, see "6.2   Sample Programs".

For information about how to monitor the operational state of a user program using the watchdog timer, see "2.7.2   Using a watchdog timer for monitoring a user program".

(2) Behavior of the WDTTO contact of the RAS external contacts interface

This sub-subsection describes the behavior of the WDTTO contact of the RAS external contacts interface under each of the following conditions.

● When the equipment starts

The WDTTO contact is closed.

● When the OS starts

The WDTTO contact is closed when the OS starts. The WDTTO contact becomes open when the watchdog timer is retriggered by the automatic retriggering feature for watchdog timer or the WdtControl function.

● When the WdtContol function is called

If dwCmd is WDT_SET:

The WDTTO contact becomes open. If the watchdog timer is not retriggered afterward within the specified timeout period, the contact becomes closed. If the watchdog timer is retriggered when the contact is closed, the WDTTO contact becomes open.

If dwCmd is WDT_STOP:

The WDTTO contact becomes open. In this case, the countdown of the watchdog timer is stopped and consequently the watchdog timer will not time out.

● When the automatic retriggering feature for watchdog timer is used

The WDTTO contact becomes open. The process for this function runs at the lowest priority (idle priority class). If the amount of CPU time used by processes with a priority higher than the idle priority class exceeds the set value, the watchdog timer times out. And the contact becomes closed. Afterward, when free CPU time becomes available and the process for this function can run, the watchdog timer resumes being retriggered and the WDTTO contact becomes open.

Figure 6-1 shows an example of the behavior of the WDTTO contact when an application controls the watchdog timer using the WdtControl function. In this example, the timeout of the watchdog timer specified by the WdtControl function is 10 seconds.
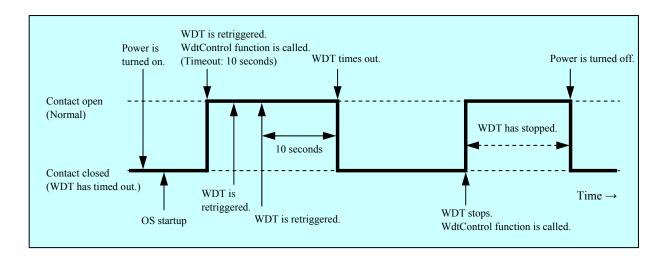
Figure 6-1   Behavior of the WDTTO Contact

Dotted lines in Figure 6-1 denote the respective states of the contact and the thick line denotes the transition of the state of the contact.

Figure 6-2 shows an example of the behavior of the WDTTO contact when the OS shuts down. In this example, the timeout of the watchdog timer is 60 seconds. The process that retriggers the watchdog timer terminates during shutdown and consequently the watchdog timer times out.
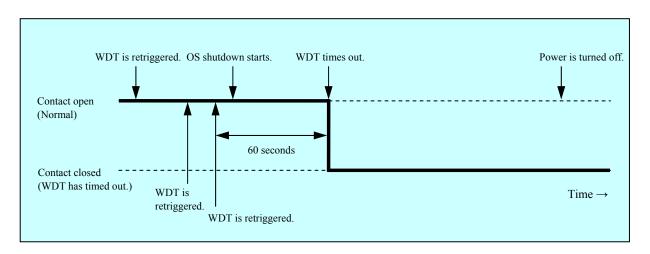
Figure 6-2   Behavior of the WDTTO Contact (When the OS Shuts Down)

Dotted lines in Figure 6-2 denote the respective states of the contact and the thick line denotes the transition of the state of the contact.

─── NOTE ──────────────────────────────────────
In the explanation above, it is assumed that the RAS external contacts interface is HJ-F2040-16. If you use HJ-F2040-17 as a RAS external contacts interface, the behavior of the contact is opposite. This contact is open when the contact in the explanation above is closed and the other way around.
───

6.1.4   Control functions for the general purpose external contact outputs
(GendoControl and GendoControlEx)
Use the GendoControl and GendoControlEx functions to control the outputs to the general
purpose external contacts of the RAS external contacts interface.

(1) Function interface (GendoControl)
  <Name>
    GendoControl - Output control for the general purpose external contact (GENDO0)

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

  <Description>
    This function performs the action specified by dwCmd on the general purpose external
    contact (GENDO0) of the RAS external contacts interface.
    This function is for backward compatibility only.
    Table 6-3 shows a list of actions that can be specified by dwCmd.

Table 6-3   List of Actions of the GendoControl Function Specified by dwCmd

dwCmd	Explanation of the action
GENDO_OPEN (0x00)	Opens the general purpose external contact (GENDO0).
GENDO_CLOSE (0x01)	Closes the general purpose external contact (GENDO0).

    If a value other than the above is specified, this function terminates with an error. If you call
    the GetLastError Windows API function, an error code W2KRAS_INVALID_PARAMETER
    is returned.

  <Diagnosis>
    If this function completes successfully, the function returns TRUE. If this function terminates
    with an error, the function returns FALSE.
    When this function terminates with an error, call the GetLastError Windows API function to
    get the error code. Error codes returned by this function on its own are as follows.

Error code (value)	Description
W2KRAS_INVALID_PARAMETER (0x2001)	There is an error in the specified parameters.
W2KRAS_NOT_INITIALIZE (0x2005)	Startup of the RAS software has not completed yet.
W2KRAS_INTERNAL_ERROR (0x2007)	An internal error has been generated.

    Other error codes come from the Windows API functions used by this function. For details
    about those error codes, refer to the Windows API help.

(2) Function interface (GendoControlEx)

<Name>

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

<Syntax>

#include <w2kras.h>

BOOL GendoControlEx(DWORD dwPort, DWORD dwCmd);

<Description>

This function performs the action specified by dwCmd on the general purpose external contact (GENDO0, GENDO1, or GENDO2) of the RAS external contacts interface specified by dwPort.

Table 6-4 shows a list of actions that can be specified by dwPort.

Table 6-4　List of Actions of the GendoControlEx Function Specified by dwPort

dwPort	Explanation of the action
GENDO0_PORT (0x01)	Performs the action on the general purpose external contact (GENDO0).
GENDO1_PORT (0x02)	Performs the action on the general purpose external contact (GENDO1).
GENDO2_PORT (0x03)	Performs the action on the general purpose external contact (GENDO2).

Table 6-5 shows a list of actions that can be specified by dwCmd.

Table 6-5　List of Actions of the GendoControlEx Function Specified by dwCmd

dwCmd	Explanation of the action
GENDO_OPEN (0x00)	Opens the general purpose external contact specified by dwPort.
GENDO_CLOSE (0x01)	Closes the general purpose external contact specified by dwPort.

If a value other than the above is specified, this function terminates with an error. If you call the GetLastError Windows API function, an error code W2KRAS_INVALID_PARAMETER is returned.

<Diagnosis>

If this function completes successfully, the function returns TRUE. If this function terminates with an error, the function returns FALSE.

When this function terminates with an error, call the GetLastError Windows API function to get the error code. Error codes returned by this function on its own are as follows.

Error code (value)	Description
W2KRAS_INVALID_PARAMETER (0x2001)	There is an error in the specified parameters.
W2KRAS_NOT_INITIALIZE (0x2005)	Startup of the RAS software has not completed yet.
W2KRAS_INTERNAL_ERROR (0x2007)	An internal error has been generated.

Other error codes come from the Windows API functions used by this function. For details about those error codes, refer to the Windows API help.

<Sample program>

We provide a sample program that uses this function in C. For information about the name of the sample program and where you can find it, see "6.2 Sample Programs".

(3) Behavior of the general purpose external contacts (GENDO0 / GENDO1 / GENDO2) of the RAS external contacts interface

Those general purpose external contacts become open when the power is turned on or off. Figure 6-3 shows the behavior of the GENDO0 contact when the GendoControl function is used.
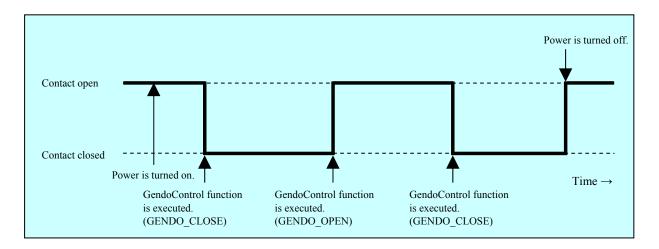

Figure 6-3   Behavior of the GENDO0 Contact

Dotted lines in Figure 6-3 denote the respective states of the contact and the thick line denotes the transition of the state of the contact.

The explanation in this sub-subsection is applicable for both RAS external contacts interfaces, HJ-F2040-16 and HJ-F2040-17.

6.1.5   Get functions for the general purpose external contact inputs (GetGendi and GetGendiEx)

Use the GetGendi and GetGendiEx functions to get the inputs for the general purpose external contacts of the RAS external contacts interface.

(1) Function interface (GetGendi)

<Name>

GetGendi - Status acquisition of the general purpose external contact input (GENDI).

<Syntax>

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

<Description>

This function gets the status of the general purpose external contact input (GENDI) of the RAS external contacts interface. This function is for backward compatibility only.

In order to use this function, in the **RAS Setup** window, clear the "**Automatically shutdown if remote shutdown is requested**" check box under **Shutdown setting**. If this check box is selected, this function terminates with an error. If you call the GetLastError Windows API function, an error code W2KRAS_RMTSTDN_ON is returned.

<Return values>

1: External contact GENDI is closed.
0: External contact GENDI is open.
0xffffffff: The function terminates with an error.

<Diagnosis>

If this function terminates with an error, the function returns 0xffffffff.

When this function terminates with an error, call the GetLastError Windows API function to get the error code. Error codes returned by this function on its own 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** check box is selected under **Shutdown setting** in the **RAS Setup** window.
W2KRAS_NOT_INITIALIZE (0x2005)	Startup of the RAS software has not completed yet.
W2KRAS_INTERNAL_ERROR (0x2007)	An internal error has been generated.

Other error codes come from the Windows API functions used by this function. For details about those error codes, refer to the Windows API help.

(2) Function interface (GetGendiEx)

<Name>

GetGendiEx - Status acquisition of the general purpose external contact inputs (GENDI, GENDI0, GENDI1, and GENDI2)

<Syntax>

#include <w2kras.h>

DWORD GetGendiEx(DWORD dwPort);

<Description>

This function gets the status of the general purpose external contact inputs (GENDI, GENDI0, GENDI1, and GENDI2) of the RAS external contacts interface.

Table 6-6 shows a list of actions that can be specified by dwPort.

Table 6-6　List of Actions of the GetGendiEx Function Specified by dwPort

dwPort	Explanation of the action
GENDI_PORT (0x00)	Gets the status of the general purpose external contact (GENDI).
GENDI0_PORT (0x01)	Gets the status of the general purpose external contact (GENDI0).
GENDI1_PORT (0x02)	Gets the status of the general purpose external contact (GENDI1).
GENDI2_PORT (0x03)	Gets the status of the general purpose external contact (GENDI2).

If a value other than the above is specified, this function terminates with an error. If you call the GetLastError Windows API function, an error code W2KRAS_INVALID_PARAMETER is returned.

If you want to specify GENDI_PORT for dwPort and get the status of GENDI, clear the "**Automatically shutdown if remote shutdown is requested**" check box under **Shutdown setting** in the **RAS Setup** window. If this check box is selected, this function terminates with an error. If you call the GetLastError Windows API function, an error code W2KRAS_RMTSTDN_ON is returned.

<Return values>

1: The external contact specified by dwPort is closed.

0: The external contact specified by dwPort is open.

0xffffffff: The function terminates with an error.

<Diagnosis>

If this function terminates with an error, the function returns 0xffffffff.

When this function terminates with an error, call the GetLastError Windows API function to get the error code. Error codes returned by this function on its own are as follows.

Error code (value)	Description
W2KRAS_INVALID_PARAMETER (0x2001)	There is an error in the specified parameters.
W2KRAS_RMTSTDN_ON (0x2003)	This function cannot get the status of GENDI because the **Automatically shutdown if remote shutdown is requested** check box is selected under **Shutdown setting** in the **RAS Setup** window.
W2KRAS_NOT_INITIALIZE (0x2005)	Startup of the RAS software has not completed yet.
W2KRAS_INTERNAL_ERROR (0x2007)	An internal error has been generated.

Other error codes come from the Windows API functions used by this function. For details about those error codes, refer to the Windows API help.

<Sample program>

We provide a sample program that uses this function in C. For information about the name of the sample program and where you can find it, see "6.2 Sample Programs".

6.1.6 Log function (MConWriteMessage)

\<Name\>

MConWriteMessage - Logging

\<Syntax\>

#include \<w2kras.h\>

VOID WINAPI MConWriteMessage(LPSTR lpBuffer)；

\<Description\>

The MConWriteMessage function writes the specified message (characters) to a log file (file name: hfwrasa.log or hfwrasb.log).

The message is written along with the time stamp.

Two log files are available and the size of each file is 64 KB. When the size of the log recorded in the currently used log file exceeds 64 KB, the log file used for logging is switched to another log file.

The parameters of this function are explained below.

lpBuffer:

This parameter specifies a pointer to a string that contains the message (characters) to be written.

In order to easily identify the application that records each log entry, specify a message that starts with the name of the application, for example.

\<Checking log information\>

This function records log information in the text format in the following files. When the size of the log recorded in the log file currently used exceeds 64 KB, the log file used for logging is switched to another log file.

・%ProgramFiles%\HFWRAS\log\hfwrasa.log

・%ProgramFiles%\HFWRAS\log\hfwrasb.log

You can check log information by opening the files above using an application such as Notepad.

The format of log information is as follows.

YYYY/MM/DD hh:mm:ss - Specified log information YYYY/MM/DD hh:mm:ss - Specified log information YYYY/MM/DD hh:mm:ss - Specified log information :	YYYY: Year (in AD) MM: Month DD: Day hh: Hour (24-hour clock) mm: Minute ss: Second

Figure 6-4　Format of Log Information

Initially, each of the above files is filled with EOF (ASCII code: 0x1a).

<Sample program>

We provide a sample program that uses this function in C. For information about the name of the sample program and where you can find it, see "6.2   Sample Programs".

NOTICE

Log function exits (asynchronously) without waiting for the data to be actually written to a log file. That means this function does not return an error even when writing to a log file fails for some reason. We recommend important information be recorded in the OS event log.

NOTE

・This function has the same name as the message console function provided by W2K-PLUS, software from Hitachi, but this function does not output to the message console.

・In order to reduce the amount of resources used, this function, for example, opens and closes a pipe every time the function is called. That means this function has relatively large overhead. Even when you record multiple lines of message, use one function call to output the message.

・This function does not support a Unicode string. Always use an ANSI string. The log entry for the message is stored in a text file. In a text file, the combination of two characters "\r\n" is recognized as a newline. If you want to include a newline in a string specified by lpbuffer, insert "\r\n" in the string.

6.1.7   Get function for the memory condition (GetMemStatus)

<Name>

   GetMemStatus - Memory status acquisition

<Syntax>

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

<Description>

   The GetMemStatus function stores the condition of the memory in this equipment to a structure
   pointed by pMemData. The parameters of this function are explained below.

   pMemData:
      This parameter specifies a pointer to a MEM_DATA structure that stores the acquired
      memory condition.

      typedef struct MEM_DATA {
         int   Dimm_Number;   //Number of DIMM slots in this equipment
         DWORD   Dimm_Status[4];   //Condition of each DIMM
      } MEM_DATA, *PMEM_DATA;

      When this function completes successfully, the maximum number of DIMMs is stored in
      Dimm_Number. This value varies depending on the model. Each element of Dimm_Status
      stores a value described in the following table. Note that the number of valid elements is
      Dimm_Number. (For example, if Dimm_Number is 2, the elements up to Dimm_Status[1]
      are valid.) The elements after that are reserved. The values of those reserved elements are
      undefined. Do not use those values.

Table 6-7   List of Values Stored in Each Element of Dimm_Status

Value	Description
MEMORY_NOMAL (0x00)	The memory is working properly.
MEMORY_ERR_DETECT (0x01)	Frequent error correction occurs.
MEMORY_NOT_MOUNTED (0x02)	DIMM is not mounted.

For this model, correspondence between elements of Dimm_Status and DMM names is as follows.

Element	DIMM name
Dimm_Status[0]	DIMM A1
Dimm_Status[1]	DIMM B1

<Diagnosis>

If this function completes successfully, the function returns TRUE. If this function terminates with an error, the function returns FALSE. When this function terminates with an error, the value stored in pMemData is invalid.

When this function terminates with an error, call the GetLastError Windows API function to get the error code. Error codes returned by this function on its own are as follows.

Error code (value)	Description
W2KRAS_INVALID_PARAMETER (0x2001)	There is an error in the specified parameters.
W2KRAS_NOT_INITIALIZE (0x2005)	Startup of the RAS software has not completed yet.
W2KRAS_MEMST_INVALID (0x2007)	The memory condition cannot be acquired.

Other error codes come from the Windows API functions used by this function. For details about those error codes, refer to the Windows API help.

<Sample program>

We provide a sample program that uses this function in C. For information about the name of the sample program and where you can find it, see "6.2 Sample Programs".

6.1.8   Get function for the failure prediction status of the drive (GetHddPredict)

\<Name\>

  GetHddPredict - Get function for the failure prediction status of the drive

\<Syntax\>

  #include \<w2kras.h\>

  BOOL GetHddPredict(PHDD_PREDICT_DATA pHddData);

\<Description\>

  This function acquires the failure prediction status of the internal drives of this equipment. This
  function is for backward compatibility only.

  The parameters of this function are explained below.

  ──── NOTE ────────────────────────────

    If the operating system is a 64-bit version, this function is not used. Use the
    function (hfwDiskStat) provided as an alternative.

---

  pHddData:

    This parameter specifies a pointer to an HDD_PREDICT_DATA structure that stores the
    acquired drive failure prediction status.

    typedef struct HDD_PREDICT_DATA {

      DWORD Hdd1_Predict;   //Failure prediction status of the drive on the drive bay1

      DWORD Hdd2_Predict;   //Failure prediction status of the drive on the drive bay2

    } HDD_PREDICT_DATA, *PHDD_PREDICT_DATA;

    Each element of the HDD_PREDICT_DATA structure above stores a value described in
    Table 6-8 when this function returns successfully.

  Table 6-8   List of Values Stored in Each Element of the HDD_PREDICT_DATA Structure

Value	Meaning
HDD_PREDICT_OFF (0x00)	The HDD is working properly.
HDD_PREDICT_ON (0x01)	Failure is anticipated.
HDD_NOT_CONNECTED (0x02)	A drive is not connected.
HDD_PREDICT_FAILURE (0x03)	Acquiring the failure prediction status failed.

<Diagnosis>

If this function completes successfully, the function returns TRUE. If this function terminates with an error, the function returns FALSE.

When this function terminates with an error, call the GetLastError Windows API function to get the error code. Error codes returned by this function on its own are as follows.

Error code (value)	Description
W2KRAS_INVALID_PARAMETER (0x2001)	There is an error in the specified parameters.
W2KRAS_NOT_INITIALIZE (0x2005)	Startup of the RAS software has not completed yet.
W2KRAS_INTERNAL_ERROR (0x2007)	An internal error has been generated.

Other error codes come from the Windows API functions used by this function. For details about those error codes, refer to the Windows API help.

6.1.9   Get function for the drive condition (hfwDiskStat)

\<Name\>

　hfwDiskStat - Drive status acquisition

\<Syntax\>

　#include　\<hfwras.h\>

　BOOL hfwDiskStat(PHFW_DISK_STATUS phfwDiskStatus);

\<Description\>

　The hfwDiskStat function stores the drive conditions to a structure pointed by phfwDiskStatus.
　The parameters of this function are explained below.

　phfwDiskStatus:

　　This parameter specifies a pointer to an HFW_DISK_STATUS structure that stores the drive
　　conditions.

　　typedef struct HFW_DISK_STATUS{

　　　DWORD　　Disk_Count;

　　　DWORD　　Disk_Status[16];　　//Drive condition

　　} HFW_DISK_STATUS, *PHFW_DISK_STATUS;

　　Disk_Count stores "2", the number of drive bays on this equipment.
　　Disk_Status[n] stores the condition of the drive on the drive bay (n + 1). The following
　　values are used to indicate a drive condition. Upper 16 bits are reserved. The values of those
　　reserved bits are undefined. Do not use those bits.

Table 6-9   List of Values Stored in Disk_Status

Defined value	Description
DISKSTAT_HEALTHY (0x00000001)	The drive is working properly.
DISKSTAT_SMART (0x00000008)	Drive failure prediction (SMART) is detected.
DISKSTAT_NOT_CONNECTED (0x00000010)	A drive is not connected.
DISKSTAT_UNKNOWN (0x00000020)	The drive condition could not be acquired.
DISKSTAT_OVERRUN (0x00000040)	The drive power-on (=used) hours exceeded the threshold. (*1)
DISKSTAT_OFFLINE (0x00000002)	The drive is disconnected from the RAID. (*2)
DISKSTAT_REBUILD (0x00000004)	A new drive is connected to the RAID and is being rebuilt. (*2)
DISKSTAT_INCOMPLETE_DATA (0x00000400)	The drive is disconnected from the RAID. (Incomplete Data) (*2)

(*1) This value is not stored if the drive usage monitoring is not enabled.
(*2) This value is returned in the B Model only.

Note that if multiple conditions exist at the same time, the logical sum of the values shown in Table 6-9 is stored.

・If drive failure prediction (SMART) is detected and the drive power-on (=used) hours exceeded the threshold at the same time:

DISKSTAT_SMART | DISKSTAT_OVERRUN (0x48) is stored.

<Diagnosis>

If this function completes successfully, the function returns TRUE. If this function terminates with an error, the function returns FALSE. When this function terminates with an error, the value stored in phfwDiskStatus is invalid.

When this function terminates with an error, call the GetLastError Windows API function to get the error code. Error codes returned by this function on its own are as follows.

Error code (value)	Description
HFWRAS_INVALID_PARAMETER (0x20000001)	There is an error in the specified parameters.
HFWRAS_NOT_INITIALIZE (0x20000002)	Startup of the RAS software has not completed yet.
HFWRAS_INTERNAL_ERROR (0x20000003)	An internal error has been generated.

Other error codes come from the Windows API functions used by this function. For details about those error codes, refer to the Windows API help.

<Sample program>

We provide a sample program that uses this function in C. For information about the name of the sample program and where you can find it, see "6.2 Sample Programs".

6.1.10   Get function for the RAID status (hfwRaidStat, RaidStat) [B Model only]

(1) hfwRaidStat
  <Name>
    hfwRaidStat - RAID status acquisition

  <Syntax>
    #include <hfwras.h>
    BOOL hfwRaidStat (PHFW_RAID_STATUS phfwRaidStatus);

  <Description of the function>
    The hfwRaidStat function stores the RAID status to a structure pointed to by phfwRaidStatus.
    The parameters of this function are explained below.

    phfwRaidStatus:
      This parameter specifies a pointer to an HFW_RAID_STATUS structure that stores the
      RAID status.

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

      Array_Count shows the number of monitored RAID arrays on this equipment (1).
      Each element of Array shows the status of the corresponding RAID. The number of valid
      elements is returned in Array_Count. (This value is 1, which means only Array[0] is valid.)
      The elements after that are reserved. The values of those reserved elements are NULL. Do
      not access those elements. The HFW_ARRAY_STATUS structure is defined as follows.

      typedef struct HFW_ARRAY_STATUS{
      DWORD   Level;   //RAID level
      DWORD   Disk_Number;   //Drives in the RAID
      DWORD   Status;   //RAID status
      DWORD   Progress;   //Progress (0 - 99[%])
      }HFW_ARRAY_STATUS, *PHFW_ARRAY_STATUS ;

Level stores the RAID level. The meaning of each value that can be stored in Level is as follows.

Table 6-10   Values Stored in Level in the HFW_ARRAY_STATUS Structure

No.	Defined value	Description
1	HFW_RAID1 (0x00000001)	RAID 1 is set up.

DiskNumber stores the value that indicates the drive bays used by the RAID. The stored value is a logical sum of some of the following.

Table 6-11   Values Stored in DiskNumber in the HFW_ARRAY_STATUS Structure

No.	Defined value	Description
1	HFW_DISK1 (0x00000001)	Drive bay1 is used by the RAID.
2	HFW_DISK2 (0x00000002)	Drive bay2 is used by the RAID.

Status stores the value that indicates the RAID status. The stored value is a logical sum of some of the following. Numbers.1 through 4 denote RAID statuses, and numbers 5 and 6 denote detailed information.

Table 6-12   Values Stored in Status in the HFW_ARRAY_STATUS Structure

No.	Defined value	Description
1	HFW_RAID_OPTIMAL (0x00000001)	The RAID is working properly.
2	HFW_RAID_DEGRADE (0x00000004)	The RAID is degenerated.
3	HFW_RAID_UNKNOWN (0x00000008)	The RAID status is unknown.
4	HFW_RAID_REBUILD (0x00010000)	One of the drives is under the rebuild process.
5	HFW_RAID_MEDIA_ERROR (0x00020000)	A media error has occurred.

For example, while the rebuild process is ongoing, the RAID status is Degraded (HFW_RAID_DEGRADE) and the detailed information is Rebuild (HFW_RAID_REBUILD). Therefore, the logical sum of those two (HFW_RAID_DEGRADE| HFW_RAID_REBUILD) is stored in Status.

Possible combinations are as follows.

Table 6-13　Possible Combinations that Can Be Stored in Status in the
HFW_ARRAY_STATUS Structure

No.	RAID status	Detailed information
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

Progress stores the progress of the rebuild process. If the RAID status value (Status) does not contain HFW_RAID_REBUILD, 0 is stored.

<Diagnosis>

If this function completes successfully, the function returns TRUE. If this function terminates with an error, the function returns FALSE. When this function terminates with an error, the value stored in phfwRaidStatus is invalid.

When this function terminates with an error, call the GetLastError Windows API function to get the error code. Error codes returned by this function on its own are as follows.

Error code (value)	Description
HFWRAS_INVALID_PARAMETER (0x20000001)	There is an error in the specified parameters.
HFWRAS_NOT_INITIALIZE (0x20000002)	Startup of the RAS software has not completed yet.
HFWRAS_INTERNAL_ERROR (0x20000003)	An internal error has been generated.
HFWRAS_NOT_RAID (0x20000101)	This equipment on which this function is being executed is not a RAID Model.

Other error codes come from the Windows API functions used by this function. For details about those error codes, refer to Windows API help.

<Sample program>

The following is a sample from a C program that uses this function. For information about the name of the sample program and where you can find it, see "6.2　Sample Programs".

(2) RaidStat

<Name>

RaidStat - RAID drive status acquisition

<Syntax>

#include   <hfwraid.h>

BOOL RaidStat(PRAID_DATA   pRaidData);

<Description of the function>

The RaidStat function stores the drive status of the RAID to a structure pointed to by pRaidData. This function is for backward compatibility only. The parameters of this function are explained below.

─── NOTE ───

If the operating system is a 64-bit version, this function is not used.
Use the function (hfwRaidStat) provided as an alternative.

pRaidData:

This parameter specifies a pointer to a RAID_DATA structure that stores the acquired drive status.

typedef struct RAID_DATA {

   DWORD Hdd1_Status;   //Drive status of drive bay1

   DWORD Hdd2_Status;   //Drive status of drive bay2

} RAID_DATA, *PRAID _DATA;

Each element of the RAID_DATA structure above stores a value described in Table 6-14 when this function returns successfully.

Table 6-14　Values Stored in Each Element of the RAID_DATA Structure

No.	Defined value	Description
1	HDD_ONLINE （0x01）	The HDD is working properly.
2	HDD_OFFLINE （0x02）	The drive is disconnected from the RAID.
3	HDD_REBUILD （0x03）	The RAID is under a rebuild (copy) process.
4	HDD_REMOVED （0x04）	A drive is not connected.
5	HDD_UNKNOWN （0x06）	Unknown status

<Diagnosis>

If this function completes successfully, the function returns TRUE. If this function terminates with an error, the function returns FALSE. When this function terminates with an error, the value stored in pRaidData is invalid.

When this function terminates with an error, call the GetLastError Windows API function to get the error code. Error codes returned by this function on its own are as follows.

Error code (value)	Description
RAID_INVALID_PARAMETER （0x2001）	There is an error in the specified arguments.
RAID_GET_STATUS_ERROR （0x2002）	An internal error has been generated.
NOT_RAID （0x2006）	This equipment on which this function is being executed is not a RAID Model.
RAID_NOT_READY （0x3001）	Startup of the RAS software has not completed yet.

Other error codes come from the Windows API functions used by this function. For details about those error codes, refer to Windows API help.

6.1.11   Status display digital LEDs control functions (SetStCode7seg, TurnOff7seg, SetMode7seg)

(1) Application status code display function (SetStCode7seg)
  <Name>
    SetStCode7seg - Displaying an application status code

  <Syntax>
    #include <ctrl7seg.h>
    BOOL SetStCode7seg(DWORD dwStCode);

  <Description>
    This function outputs an application status code on the status display digital LEDs. On the
    status display digital LEDs, the value specified by this function is displayed in hexadecimal.

    The parameters of this function are explained below.

    dwStCode:
      This parameter specifies an application status code to be displayed on the LEDs. You can
      specify a value between 0 and 255. If a value other than the above is specified, this function
      terminates with an error. If you call the GetLastError Windows API function, an error code
      CTRL7SEG_INVALID_PARAMETER is returned.

  <Diagnosis>
    If this function completes successfully, the function returns TRUE as a return value. If this
    function terminates with an error, the function returns FALSE.
    When this function terminates with an error, call the GetLastError Windows API function to
    get the error code. Error codes returned by this function on its own are as follows.

Error code (value)	Description
CTRL7SEG_INVALID_PARAMETER (0x2001)	There is an error in the specified parameters.

    Other error codes come from the Windows API functions used by this function. For details
    about those error codes, refer to the Windows API help.

  <Sample program>
    We provide a sample program that uses this function in C. For information about the name of
    the sample program and where you can find it, see "6.2   Sample Programs".

(2) Application status code clear function (TurnOff7seg)

<Name>

TurnOff7seg - Turning off an application status code

<Syntax>

#include <ctrl7seg.h>

BOOL TurnOff7seg(VOID);

<Description>

This function clears the application status code currently displayed on the status display digital LEDs. When this function is called, the status display digital LEDs are turned off.

<Diagnosis>

If this function completes successfully, the function returns TRUE. If this function terminates with an error, the function returns FALSE.

When this function terminates with an error, call the GetLastError Windows API function to get the error code. The error code comes from the Windows API functions used by this function. For details about those error codes, refer to the Windows API help.

<Sample program>

We provide a sample program that uses this function in C. For information about the name of the sample program and where you can find it, see "6.2　Sample Programs".

(3) Status display mode setup function (SetMode7seg)

<Name>

SetMode7seg - Setting up the status display mode

<Syntax>

#include <ctrl7seg.h>

BOOL SetMode7seg(DWORD dwMode);

<Description>

This function configures the status display mode of the status display digital LEDs.

The parameters of this function are explained below.

dwMode:

This parameter specifies the "status display mode" of the status display digital LEDs.

Table 6-15 shows a list of values that can be specified by dwMode.

Table 6-15　List of Values Specified by dwMode in the SetMode7seg Function

dwMode	Explanation of the action
APPST_MODE (0x00)	Selects the application status display mode.
RASST_MODE (0x01)	Selects the status display mode.

If a value other than the above is specified, this function terminates with an error. If you call the GetLastError Windows API function, an error code CTRL7SEG_INVALID_PARAMETER is returned.

<Diagnosis>

If this function completes successfully, the function returns TRUE. If this function terminates with an error, the function returns FALSE.

When this function terminates with an error, call the GetLastError Windows API function to get the error code. Error codes returned by this function on its own are as follows.

Error code (value)	Description
CTRL7SEG_INVALID_PARAMETER (0x2001)	There is an error in the specified parameters.

Other error codes come from the Windows API functions used by this function. For details about those error codes, refer to the Windows API help.

<Sample program>

We provide a sample program that uses this function in C. For information about the name of the sample program and where you can find it, see "6.2　Sample Programs".

6.2 Sample Programs

Sample program file in C that use RAS library functions are stored in the %ProgramFiles% \HFWRAS\sample directory. Use those files for reference when you develop a program or check the operation of those functions.

The following table shows a list of sample programs.

Table 6-16　List of Provided Sample Programs

No.	File name	Description
1	shutd.c	A sample program for the BSSysshut function
2	wdt.c	A sample program for the WdtControl function
3	gendoex.c	A sample program for the GendoControlEx function
4	gendiex.c	A sample program for the GetGendiEx function
5	MCon.c	A sample program for the MconWriteMessage function
6	MemErr.c	A sample program for the GetMemStatus function
7	hfwDiskStat.c	A sample program for the hfwDiskStat function
8	7seg.c	A sample program for the status display digital LEDs control functions (SetStCode7seg, TurnOff7seg, and SetMode7seg)
9	FanErr.c	A sample program for detecting a RAS event (For information about the RAS event notification, see "4.2　RAS Event Notification".)
10	hfwRaidStat.c	A sample program that uses the hfwRaidStat function

CHAPTER 7   FEATURES RELATED TO MAINTENANCE AND FAILURE ANALYSIS

7.1   Notifying the Cause of the STOP Error Code

7.1.1   Overview

This equipment records the memory contents in memory dump files when the system is forcibly recovered from OS lockup, an NMI is generated due to hardware failure, or uncorrectable memory error occurs. At the same time, a blue screen appears and the STOP error code 0x00000080 (hereinafter denoted as 0x80) is displayed.

The STOP error code 0x80 has multiple possible causes. You may be able to identify the cause based on the information displayed under the STOP error code, or at least you may be able to clarify which viewpoint to use for cause analysis based on that information. This function detects a blue screen caused by an error with the STOP error code 0x80 and records the cause of the blue screen in the eventlog after restart this equipment.

─── NOTE ───

- These causes may not be recordable in the event log if you perform an action (for example, shutdown) that aborts the process of the STOP error code cause analysis.
  In the case of a 4GB memory dump file, analysis of the STOP error code takes about three minutes.
- In Windows® 10, the STOP error code **0x80** and the detailed description of the error are not displayed on a blue screen. Only an identifier (NMI_HARDWARE_FAILURE) is displayed.

7.1.2   Supported causes of the STOP error code
This function is activated when a blue screen appears due to a cause described in Table 7-1. Other STOP errors of Windows® are not supported.

Table 7-1   Supported STOP Error Causes

Cause	STOP error code (*1)	Contents of the STOP error screen (*1)
An NMI is generated by the reset button or reset signal is input to the external contact RMTRESET.	0x00000080	Hardware malfunction. == Detailed Information == 0x9201:IOCHK Error.
An NMI is generated due to parity error at the PCI bus, LPC bus, or others	0x00000080	Hardware malfunction. == Detailed Information == 0x9202:PCI Bus Parity Error.
Uncorrectable memory error occurs	0x00000080	Hardware malfunction. == Detailed Information == 0x????: XXXXXXXXX (*2)

(*1) In Windows® 10, these items are not displayed on a blue screen.
(*2) Either one of the following is displayed depending on the error cause.
   0x9217: Uncorrectable Error at DIMM B1.
   0x9218: Uncorrectable Error at DIMM A1.

For information about what actions you should take when a blue screen appears, refer to the "HF-W2000 Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".

7.1.3   Event log

Table 7-2 shows a list of event log entries this function records. Those event log entries are recorded in the system log.

Table 7-2   Event Log Entries Recorded by This Function

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

(Note) When an event log entry is recorded, the %1 above contains a message, and the %2 above contains the code corresponding to the %1.

%1	%2
Reset signal was input.	0x9201
PCI Bus Parity Error	0x9202
Uncorrectable Error at DIMM B1	0x9217
Uncorrectable Error at DIMM A1	0x9218

7.2 Log Information Collection Window

7.2.1 Overview

In the log information collection window, you can take the following actions by using a graphical user interface.

(1) Collecting log data

This function saves the data used for preventive maintenance and post-failure analysis of the problem. The data is compressed and saved as one file (File name: logsave.zip).

(2) Collecting memory dump files

This function collects the memory dump files saved by the OS. The data is saved as a compressed file (File name: memory.zip). At the same time, minimum memory dump files are also collected.

NOTICE

CPU load increases while memory dump files are being collected. While CPU load is high, operation of user applications can be disturbed. Make sure that you do not collect memory dump files using the log information collection window while the applications for business use are running on this equipment.

7.2.2 Starting the log information collection window

To start the log information collection window, follow the procedure below:

You need to have administrator privileges to use this window. Log on to the computer using an administrator account and start the window.

1. Click **Start**.
2. Point to **All Programs** (or **All Apps**) > **RAS Software**.
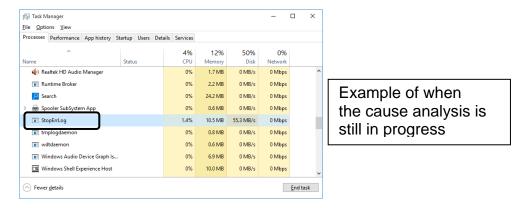3. Click **RAS Maintenance Support**.

NOTE

- The log information collection window cannot be used by multiple users at the same time. If you use, for example, user switching to try to start instances of the log information collection window from multiple consoles, an error may occur. If an error occurs, close the log information collection window on another console, and then start the window.
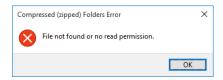

NOTE

If you are using Windows®10, do not collect a memory dump file immediately after the OS starts after the blue screen is displayed. When the OS restarts after the blue screen is displayed, the memory dump file is accessed by the RAS software to analyze the cause of the STOP error code, and a memory dump cannot be collected during this period. Wait for a while, and then collect a memory dump file. (About three minutes in the case of HF-W2000 Model 45 A Model with 4GB memory) <u>Collect a memory dump file after the STOP error code cause analysis is complete.</u> You can check whether the STOP error code cause analysis is complete using the procedure described below.

[Check procedure for whether the STOP error code cause analysis is complete]
1. Start Task Manager.
2. Click **More details** in the Task Manager window and select **Processes** tab.
3. Check if the **StopErrLog** process is running. When the **StopErrLog** process is not running, STOP error cause analysis is finished.

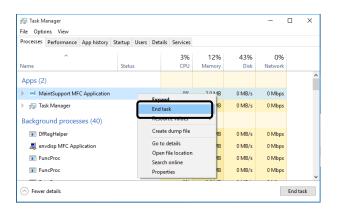
Example of when the cause analysis is still in progress

If you collect a memory dump file during the STOP error code cause analysis, the following dialog box is displayed, and compression of the memory dump is stopped. If the following dialog box is displayed, follow the procedure below to close the log information collection window.

[Stop procedure for the log information collection window]
1. Start Task Manager.
2. Click **More details** in the Task Manager window and select **Processes** tab.
3. Right-click **MaintSupport MFC Application**.
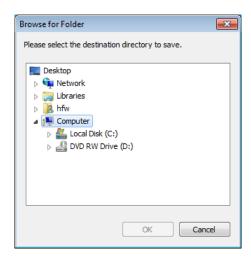4. Select **End task**, then terminate **MaintSupport MFC Application**.

7.2.3   Using the log information collection window

1. The log information collection window appears. By default, both the "**Gather log data**" and "**Gather memory dump files**" check boxes are selected. If you do not need one of those two, clear the check box for the one you do not need, and then click **Continue**.

2. If the "**Gather memory dump files**" check box is selected, you will receive the following message. Click **OK**. If you click **Cancel**, you go back to the log information collection window without executing the maintenance operation.
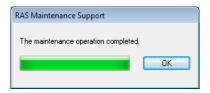

3. The following dialog box is displayed. Specify the destination directory to save and click **OK**. If you want to cancel the maintenance operation, click **Cancel**. If you click **Cancel**, you go back to the log information collection window without executing the maintenance operation.

4. The information selected at step 1 is collected. During the process, a window is displayed to show the progress. If the process finishes successfully, the following window appears.
Do not do anything on the windows that appear during the process, and wait until the following window appears.

5. A directory is created under the directory specified as the destination directory to save. The name of the directory is assigned based on the date and time of the operation. Under the directory just created, collected data is saved.
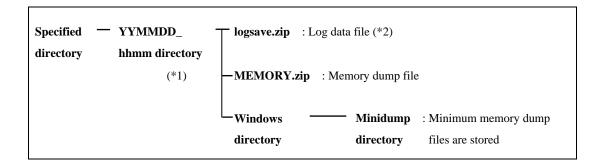
Figure 7-1   Organization of the Folders for Collected Data

(*1) The name of the directory is "YYMMDD_hhmm".

YY: Lower two digits of the year, MM: Month, DD: Day, hh: Hour, mm: Minute

Example: If the date is saved at 13:59 on Jan 1, 2014

Directory name: "140101_1359"

(*2) The following data is saved.

・If **Gather log data** is selected: logsave.zip

・If **Gather memory dump files** are selected: MEMORY.zip and minimum memory dump files

7.2.4   Finishing the log information collection window

To close the log information collection window, click **Cancel.**

7.3   Logging the Trend of the Temperature inside the Chassis

7.3.1   Overview

This function periodically measures the temperature inside the chassis of this equipment and records the data into a log file. You can adjust the logging interval for the temperature inside the chassis by using the logging interval setup command. The default setting for the logging interval is 60 minutes. You can select the setting from three options: 10, 30, and 60 minutes.

7.3.2   Log files

The temperature inside the chassis is recorded in a log file at the specified logging intervals. When this equipment runs continuously for 8 hours or more, the highest and lowest temperatures in 8 hours are also recorded every 8 hours. For either file, if the log gets full, log entries are overwritten from the oldest entry.
Table 7-8 shows the name of each log file.

Table 7-3   Log Files

Parent folder	File name	Description
%ProgramFiles%\HFWRAS\log	temp.csv	The temperature inside the chassis is recorded in this file at the logging intervals. (Maximum of 51200 entries)
	temp_mm.csv	The highest and lowest temperatures in 8 hours are recorded in this file. (Maximum of 1100 entries)

<Checking log information>
You can check log information by opening the log files using an application such as Notepad. The files are in the csv format. You can use spreadsheet or database software to load the log information and draw graphs.
You can also use the logsave command to collect those log files. For information about how to use the logsave command, refer to "7.5.1   Log information collection command (logsave)" in the "HF-W2000 Model 48/45 INSTRUCTION MANUAL (manual number WIN-62-0066)".

<Log information format>

The format of log information is as follows.

(1) temp.csv

```
YYYY/MM/DD hh:mm:ss, yxxx

          :

          :
```

YYYY: Year, MM: Month, DD: Day, hh: hour (24-hour clock), mm: minute, ss: second,

y: Sign (+ or -), xxx: (Temperature (°C))

If acquiring the temperature fails, xxx is replaced with "---".

Figure 7-2   Format of Log Information 1

(2) temp_mm.csv

```
YYYY/MM/DD hh:mm:ss, yxxx, yzzz

          :

          :
```

YYYY: Year, MM: Month, DD: Day, hh: hour (24-hour clock), mm: minute, ss: second,

y: Sign (+ or -), xxx: (Highest temperature (°C))

y: Sign (+ or -), zzz: (Lowest temperature (°C))

Figure 7-3   Format of Log Information 2

7.3.3　Logging interval setup command

<Name>

tmplogset - Setting up the logging interval

<Syntax>

tmplogset

<Function>

This command configures the interval for logging the trend of the temperature inside the chassis.

The following shows how to use the command.

1. Start Command Prompt.

You need to have administrator privileges to run this command. Log on to the computer as an administrator account and start Command Prompt.

2. At the command prompt, run the tmplogset command. The following initial screen is displayed along with the current setting. If you type 2 in the initial screen, the tmplogset command exits without changing the setting.

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

3. If you type 1 and then press **Enter**, the following message is displayed.

```
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. Type the number corresponding to the new interval you want to select and press **Enter**. If the number you typed is out of range, the following message is displayed to prompt you to type a correct number.

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

5. If you type a number between 1 and 3, the following message is displayed. The "×" below denotes the time you selected.

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

6. If you press **y** and then press **Enter**, the logging interval is updated to the new value and the command exits. The new setting becomes effective when the command exits. If you want to confirm the change, run this command again and check the initial screen.
If you do not want to change the logging interval, press **n** and press **Enter**. The following message is displayed and the command exits without changing the setting.

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

If you do not have administrator privileges when you run the command, the following message is displayed and the command exits.

```
>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 when you run the command, the following message is displayed and the command exits.

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

In the message above, "API Name" actually shows the name of the Windows API with the error. "Error Code" shows the error code in hexadecimal. If this message is displayed, run the command again.

CHAPTER 8　SIMULATING THE HARDWARE STATUS

8.1　Hardware Status Simulation

8.1.1　Overview

This function simulates the hardware status of this equipment. By simulating the hardware status, you can test a user application and check the notification interface of the RAS software without actual hardware failure.

When the hardware status is simulated, the RAS software is set to a special mode called "simulation mode". In "simulation mode", monitoring of the actual hardware status is disabled. You must never use this equipment in simulation mode for business use.

This function simulates the following hardware conditions.

・Fan condition
・Temperature condition inside the chassis
・Drive condition (including drive failure prediction (SMART monitoring) condition and whether the used hours exceeded the threshold)
・Memory condition
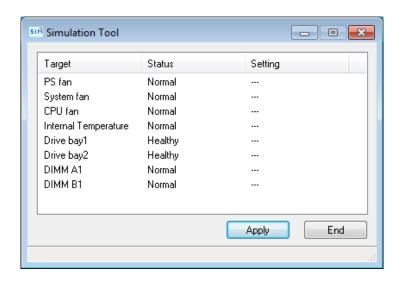・RAID status (B Model only)

Figure 8-1　Simulation Tool Window

The monitoring function of the RAS software detects change in the simulated hardware status and notifies the change through various interfaces.

For information about the interfaces used for notification, see the following sections in this manual.

・Fan condition: "2.1   Fan Monitoring"
・Temperature condition inside the chassis:
                         "2.2   Monitoring Temperature inside the Chassis"
・Drive condition: "2.3   Drive Failure Prediction Function (SMART Monitoring)"
                         and "2.4   Drive Usage Monitoring"
・Memory condition: "2.5   Memory Monitoring"
・RAID status：   "2.8   RAID Monitoring [B Model only]"

NOTICE

While the equipment is running in simulation mode, monitoring of the actual hardware status is disabled. Errors including fan failure and abnormal temperature cannot be detected. You must never use this equipment in simulation mode for business use. Use the simulation function only for testing a user application and checking the notification interface of the RAS software.

───── NOTE ─────

・While the equipment is running in simulation mode, the following functions cannot be used:
  1. OS Hung-up Monitoring
・In simulation mode, the memory monitoring function records an event (Event ID: 525) in the event log only when memory error is detected for the first time. Afterward, even if memory error persists, the memory monitoring function does not record events in the event log.
・In simulation mode, when the drive failure prediction function records an event (Event ID: 265) of SMART detection in the event log, a string "XXXXXXX" is used for the model name of the drive.
  Also note that you cannot simulate a situation that the drive condition is "Unknown" or multiple conditions (for example, SMART detection and used hours exceeded) occur at the same time.
・RAID status is automatically updated based on the condition of the drives. Note that you cannot simulate a situation in which the RAID status is "Unknown" or a situation in which media errors occur.

8.1.2   Using the simulation function

Run the simulation mode start command at the command prompt to set the RAS software
to the "simulation mode". When the RAS software transitions to the "simulation mode",
the **Simulation Tool** window appears on the screen.

You can use this window to simulate the condition of hardware devices. Note that in order
to exit the simulation mode, you need to restart this equipment.

This subsection explains the procedure of using the simulation mode.

(1) Overview of the procedure of using the simulation function

The figure below shows a rough flow chart of using this function. From when the
simulation mode start command is executed until when OS shutdown completes, the
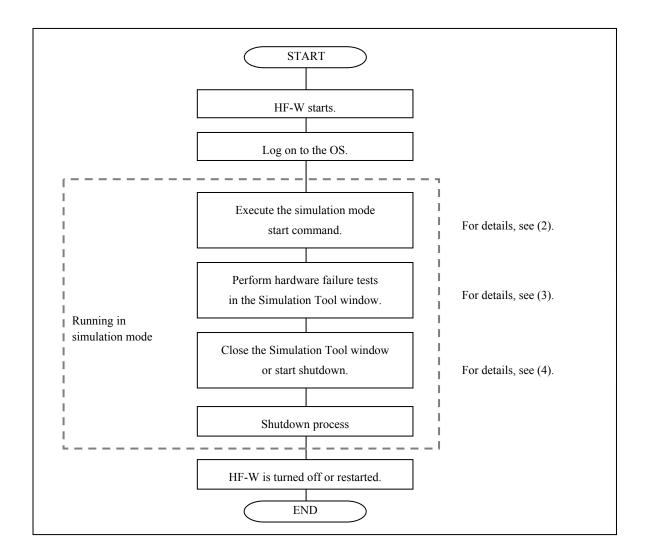RAS software runs in "simulation mode".

Figure 8-2   Procedure to Use the Simulation Mode
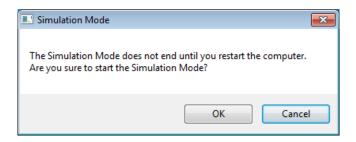
(2) Starting the simulation mode

To start the simulation mode, run the simulation mode start command (simrasstart command) at the command prompt.
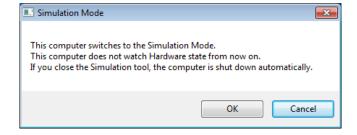
─── NOTE ───

・The simulation mode cannot be started from a remote desktop. Before you start the simulation mode, other logged-on users must log off.
・If the RAS software has already detected hardware failure, the simulation mode cannot be started. Remove the cause of the failure before using the simulation mode.

1. Start Command Prompt.

You need to have administrator privileges to run the simulation mode start command. Log on to the computer using an administrator account and start Command Prompt.

2. At the command prompt, type the following, and then press Enter.

\%SystemDrive%\"Program Files"\HFWRAS\sim\simrasstart

3. You will receive the following message about exiting the simulation mode. Click **OK**. If you click **Cancel**, the simulation mode does not start.

```
Simulation Mode                                    ✕

The Simulation Mode does not end until you restart the computer.
Are you sure to start the Simulation Mode?

                              OK          Cancel
```

4. You will receive the following message to notify that the simulation mode starts. Click **OK**. If you click **Cancel**, the simulation mode does not start.

```
Simulation Mode                                    ✕

This computer switches to the Simulation Mode.
This computer does not watch Hardware state from now on.
If you close the Simulation tool, the computer is shut down automatically.

                              OK          Cancel
```

5. The **Simulation Tool** window appears. After this point on, this equipment runs in simulation mode. Monitoring hardware failure is now disabled.

───── NOTE ─────

The simulation function performs the following actions while running in simulation mode.
・The status indicator is lit between green and red. The CPUSTOP contact switches back and forth between OPEN and CLOSE.
・Every 10 seconds, the Windows® Exclamation sound is played twice. (Only when speakers are connected)

(3) Using the Simulation Tool window

When the RAS software transitions to the simulation mode, the **Simulation Tool** window appears as shown in the following figure.

You can use the **Simulation Tool** window to change the condition of hardware devices.

When the **Simulation Tool** window starts, all hardware devices are set to the normal status.
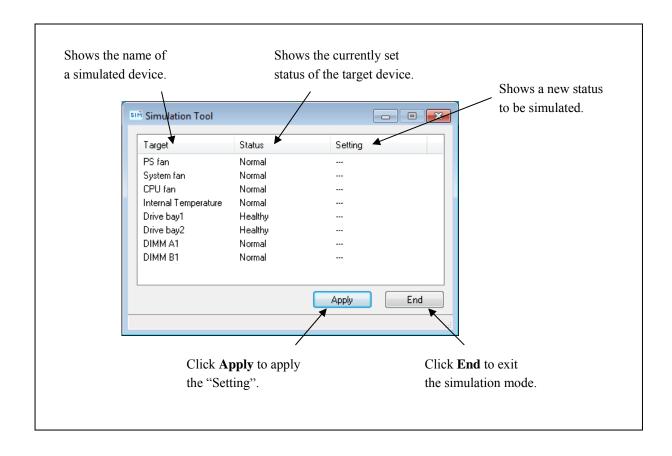
Shows the name of a simulated device.

Shows the currently set status of the target device.

Shows a new status to be simulated.

Click **Apply** to apply the "Setting".

Click **End** to exit the simulation mode.

Figure 8-3   Description of Each Part in the Simulation Tool Window

● Target

Shows the name of each simulated hardware device.

Category	Target
Fan condition	PS fan, System fan, CPU fan
Temperature condition inside the chassis	Internal Temperature
Drive condition	Drive bay1, Drive bay2
Memory condition	DIMM A1, DIMM B1

● Status

Shows the currently set status of each simulated hardware device. The following shows a list of statuses for each hardware device.

Target	Status
PS fan, System fan, CPU fan	Normal, Error
Internal Temperature	Normal, Error
Drive bay1, Drive bay2	Healthy, SMART Detected, Overrun, Not Connected, Offline (*), Rebuild (*), Incomplete Data(*)
DIMM A1, DIMM B1	Normal, Error, Not Mounted

(*) B Model only

After the **Simulation Tool** window starts, all hardware devices are set to "Normal" or "Healthy".

● Setting

Shows a new status to be simulated for each target hardware device.
If no status to be simulated is set, "---" is displayed. (After the **Simulation Tool** window starts, "---" is displayed for all hardware devices.)

● Apply button

If you click this button, all "Setting" are applied to the "Status" of hardware devices.
The monitoring function of the RAS software detects change in the "Status" of the hardware devices and notifies the change through various interfaces.

──── NOTE ────

A new hardware status is applied to the notification interface of the RAS software when the following time elapses after you click **Apply** in the **Simulation Tool** window. Wait for the time specified below before you check the result of simulation.

・Fan condition: About 10 seconds later
・Temperature condition inside the chassis: About 15 seconds later
・Drive condition: About 5 seconds later
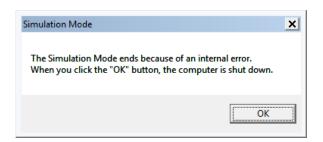・Memory condition: About 10 seconds later

● End button

If you click this button, shutdown is executed to exit the simulation mode. Before shutdown is executed, the following message is displayed. Save the data, for example, and then click **OK**. If you click **Cancel**, the **Simulation Tool** window does not exit.
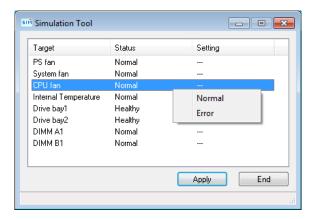

NOTE

Even if the **Simulation Tool** window exits due to an internal error or the like, shutdown is executed automatically to exit the simulation mode. Before shutdown is executed, the following message is displayed. Save the data, for example, and then click **OK**.
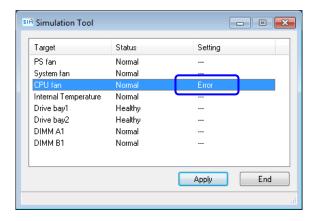

● Minimize button ([_] button)

Click the **Minimize** button at the upper right corner of the **Simulation Tool** window to minimize the **Simulation Tool** window.

● Close button ([×] button)

Click the **Close** button at the upper right corner of the **Simulation Tool** window to execute shutdown and exit the simulation mode. The behavior after this button is clicked is the same as when you click **End**.

The following procedure shows how to simulate a hardware status using the **Simulation Tool** window.
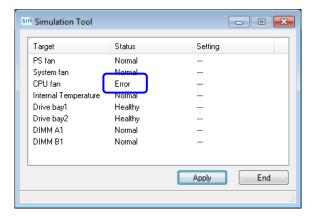
1. Right-click a hardware item you want to simulate. A popup menu is displayed. The menu lists the statuses you can select based on the current hardware status.

2. If you select a status you want to simulate on the popup menu, the selected status is displayed in "Setting".

3. To apply the status displayed in "Setting" to the hardware status, click **Apply**. As a result, the "Status" in the **Simulation Tool** window is updated.

NOTE

If the "Setting" is not selected (that is, "---" is shown) or is the same as the "Status" when you click **Apply**, the current "Status" does not change.

The following shows a list of items on the popup menu that are displayed when you right-click each hardware item.

On the popup menu, the current status and the statuses that the current status can transition to are displayed. Note that if the current status is the only status that can be displayed on the menu, "None" is displayed shaded on the menu.

● Fan

No.	Current status	Statuses in the popup menu	Note
1	Normal	Normal, Error	
2	Error		

● Internal Temperature

No.	Current status	Statuses in the popup menu	Note
1	Normal	Normal, Error	
2	Error		

● Drive

<A/S Model>

No.	Current status	Statuses in the popup menu	Note
1	Healthy	Healthy, SMART Detected, Overrun, Not Connected	(*1)
2	SMART Detected	Healthy, SMART Detected, Not Connected	(*1) (*2)
3	Overrun	Healthy, Overrun, Not Connected	(*1) (*2)
4	Not Connected	Healthy, Not Connected	(*1) (*3)

(*1) The equipment never starts without drive bay1 being mounted, and consequently, "Not connected" is not displayed for drive bay1 on the popup menu.

(*2) Transition from "SMART Detected" or "Overrun" to "Healthy" means that the target drive is replaced with a new one and that the new drive is connected.

(*3) We suppose transition from "Not Connected" means a new drive is connected. Consequently, transition to "SMART Detected" or "Overrun" is not allowed.

<B Model>

The following shows the list of items on the menu when you right-click Drive bay1 or Drive bay2.

No.	Current status	Statuses in the popup menu	Note
1	Healthy	Healthy, SMART Detected, Overrun, Not Connected, Offline, Incomplete Data	(*1)
2	SMART Detected	SMART Detected, Not Connected, Offline, Incomplete Data	(*2)
3	Overrun	Overrun, Not Connected, Offline, Incomplete Data	(*2)
4	Not Connected	Not Connected, Rebuild	
5	Offline	Not Connected, Offline, Rebuild	
6	Rebuild	Healthy, Not Connected, Offline, Rebuild	(*3) (*4)
7	Incomplete Data	Healthy, Not Connected, Offline, Rebuild, Incomplete Data	(*3) (*4)

(*1) If the RAID status is "Degrade", "Offline" and "Not Connected" are not displayed.

(*2) If the RAID status is "Degrade", "None" is the only item on the menu and is grayed out.

(*3) Transition of the status from "Rebuild" to "Healthy" means the rebuild process is complete.

(*4) We suppose transition from "Rebuild" means a new drive is connected. Consequently, transition to "SMART Detected" or "Overrun" is not allowed.

● Memory

No.	Current status	Statuses in the popup menu	Note
1	Normal	Normal, Error, Not Mounted	(*)
2	Error		(*)
3	Not Mounted		(*)

(*) In the case of this equipment, the equipment never starts with the DIMM A1 not being mounted, and consequently, "Not Mounted" is not displayed for the DIMM A1 on the popup menu.

(4) Exiting the simulation mode

To exit the simulation mode, you need to restart this equipment.

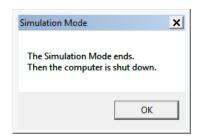For this reason, this equipment is automatically shut down in order to exit the simulation mode.

There is no limitation for how the equipment can restart in order to exit the simulation mode. In the same ways as normal mode, this equipment shutdown in the following cases(reasons). Restart the equipment after shutdown.

・ Shutdown is executed on the **Start** menu.
・ A system shutdown API function such as the BSSysshut and ExitWindowsEx functions is executed.
・ The equipment is automatically shut down because fan failure, abnormal temperature inside the chassis, or a remote shutdown signal input is detected.
・ Ctrl+Alt+Delete is pressed and using the power button at the lower right corner of the screen, the equipment is shut down.
・ A blue screen appears because, for example, the reset button is pressed.
・ The power is forcibly turned off using the power switch.
・ The **End** button or the **Close** button [×] in the **Simulation Tool** window is clicked.
・ The **Simulation Tool** window is terminated with an error.

──── NOTE ────

・ As explained above, the simulation mode exits when the equipment restarts after automatic shutdown caused by simulated fan failure or abnormal temperature inside the chassis. To start the simulation mode again, execute the simulation mode start command.
When you execute shutdown or logoff, the following message is displayed to notify that the simulation mode will exit and that the system will shut down.
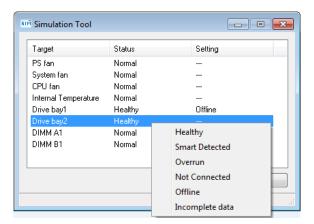

・ When a remote connection is used, shutdown process can be delayed when shutdown is executed due to a reason listed above.

8.1.3   Precautions when you use the Simulation Tool window

(1) When the new status to be simulated is finalized
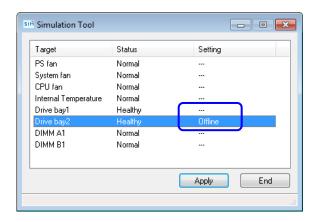From when you select a new status to be simulated from a popup menu on the **Simulation Tool** window until when you click **Apply**, the new status to be simulated is not finalized and you can change it. The actual status used for simulation is the one shown in "Setting" in the **Simulation Tool** window when you click **Apply**.

(2) Simulating the status of the drives in the RAID
In the case of B Model, the RAID status is determined by the simulation of the condition of the drives used in the RAID. In some cases, if you select a new status to be simulated for both drives, the condition of the first drive you selected may change depending on the condition of the second drive you selected. The following example illustrates the case where "Offline" is selected for both drives.

   1. Select **Offline** for drive bay1. Then, select **Offline** for drive bay2. (If you select **Not Connected** instead, the result will be the same.)

   2. When **Offline** is selected for drive bay2, the condition of the drive bay1 is changed back to "---" (Not selected).

8.1.4 Event log entries

In order to clearly show which log entries for hardware failure originate from the simulation function, this function records event log entries listed in the following table. Note that a log entry with Event ID 252 is recorded at the timing when you click **Apply** in the **Simulation Tool** window. This log entry is recorded even when all "Settings" are not specified.

Table 8-1　Event Log Entries Recorded by This Function

Event ID	Source	Type	Category	Description
250	HFWSIM_SYS	Information	HFWSIM	The Simulation Mode started.
251	HFWSIM_SYS	Information	HFWSIM	The Simulation Mode ended.
252	HFWSIM_SYS	Information	HFWSIM	The following conditions were set at the Simulation Mode. 　PS fan　　　　　　　: %1 　System fan　　　　　: %2 　CPU fan　　　　　　: %3 　Internal Temperature : %4 　Drive bay1　　　　　: %5 　Drive bay2　　　　　: %6 　DIMM A1　　　　　　: %7 　DIMM B1　　　　　　: %8

The %x above denotes the corresponding setting in the **Simulation Tool** window.

8.1.5 Remote notification

This function notifies of the transition of the RAS software to simulation mode using trap notification so that an SNMP manager that monitors this equipment in a remote location can know that the hardware statuses it acquires and the trap notifications for hardware failure (and hardware recovery) it receives are generated in simulation mode. In addition, the object value for the operational mode of the RAS software is changed to the one for the simulation mode.

― NOTE ―

The contents of the hardware status that can be acquired and a trap notification that can be notified are the same as in the normal mode. For details about the objects of the extended MIB for HF-W, see "4.5　Remote Notification".