Wii Programming Guidelines

Version: 1.08

2008/04/02

The content of this document is highly confidential
and should be handled accordingly.

Table of Contents

Tables

Revision History

Version	Revision Date	Description
1.08	2008/04/02	◆ Standardized the notation regarding inquiries to Nintendo. ◆ Standardized the leads for each section, noting that the items in the section are to be supported. ◆ Standardized the names for the Strap Usage Screen, the Disc Channel and the Channel Banner screen. ◆ Chapter 2 All The following sections were added. 2.21 Terminology and Image 2.21.1 Standardized Terminology [Required] 2.21.2 Restrictions on Images Used Instead of Names [Recommended] 2.22 Using the Time from the Wii System Clock 2.22.1 Procedure when Time of Wii System Clock Differs Greatly from Real Time [Required] 2.22.2 Prompting a User to Check the Current Time [Recommended] ◆ Deleted mention of Nintendo GameCube Memory Card from section 2.1 Prohibition of Dependency on Device Specifications [Required] ◆ Updated the contents of section 2.3 Startup Time Restriction [Recommended] to clearly state that the restriction does not include the 100 frames necessary for switching from progressive to interlaced display, or from EURGB60 to PAL display. ◆ Updated section 2.6 Using the Latest Packages [Recommended], changing the reference to library and tool versions to read "latest package." Clearly stated that this involves the Revolution SDK and its patches and related tools distributed by Nintendo. ◆ Rewrote section 2.7 Removal of Debug Information [Required] so it is easier to understand that caution is needed when creating the master data. Restructured the paragraph order so that the flow of text is easier to understand. ◆ Updated the content of section 2.8 Prohibition of Device Access Without the Use of a Library [Required] to talk about access to devices via OS and the Revolution SDK. ◆ Deleted mention of Nintendo GameCube Memory Card from section 2.16 Program Distribution Method [Required] and added mention of signature and authentication as measure against alteration of loaded data. ◆ Added links in section 2.18 [North American, European, and Australian Versions] License Display of Licensee Titles [Required] for banners that display "Licensed by Nintendo" and for the Strap Usage screen. ◆ Updated the terminology in section 2.19 Using Firmware of Specified Versions [Required], changing "network-compatible firmware" to "extension firmware, and changing "non-network-compatible firmware" to "standard firmware." ◆ Added mention to consider size of DOL file in example of `OSExec*` function in section 2.20 Time Restriction when Screen Is in Static State [Required]. Also added sound as means of verifying user operation of Controller, and added the request that Nintendo be contacted if display needs to be in static state for longer than 10 seconds. ◆ For easier comprehension, rearranged section 3.3 Prohibition of Sustained Continuous Non-Sequential Access [Required]. Added criteria for prolonged, continuous non-sequential access. ◆ Added the following items to section 3.7 Data Layout for Double Layer Disc: - 3.7.1 Avoid Disc Access Spanning Layers [Required] - 3.7.2 Avoid Reading Data to Alternate Layers [Recommended] ◆ Standardized subheads and content in Chapter 6 Video in the following ways: - Added the word "display" to references to progressive and interlaced. - Updated content to clarify that display can be switched from progressive display to interlaced display, and from EU RGB 60 display to PAL display. ◆ Changed the title of section 6.5 Procedure for Switching from Progressive Display to Interlaced Display [Recommended]. Previously it read "Interlaced/Progressive." Also updated the content to read more clearly. ◆ Deleted section 6.6 Addition of Interlaced/Progressive Switch Operation because there is no operation switching display mode from interlaced to progressive. ◆ Deleted section 6.7 Reset Operation During Progressive Output because the information duplicates the content in section 6.5 Procedure for Switching from

Version	Revision Date	Description
		Progressive Display to Interlaced Display [Recommended].
		◆ Added the word "display" to the text of Changed section 6.8 Televisions that Support Progressive Output [Recommended].
		◆ Changed the title of 6.11 Switching from EURGB60 to PAL [Recommended]. Previously it read PAL/EU RGB 60. Also updated the content to read more clearly.
		◆ Deleted section 6.12 Addition of Interlaced/Progressive Switch Operation because there is no operation switching display mode from PAL to EU RGB 60.
		◆ Deleted section 6.13 Reset Operation During EU RGB Output because the information duplicates the content in 6.11 Switching from EURGB60 to PAL [Recommended].
		◆ Deleted section 6.23 Maintaining Appropriate Video Output because it is checked by the library and it is not necessary for the application to support it.
		◆ Rewrote the example of letterboxing in section 6.18 Support for 4:3 Aspect Ratio [Required] for easier understanding.
		◆ Deleted section 6.20 Check Before Switching to Progressive Display because it is checked with the `VIGetScanMode` function.
		◆ Added an example to section 6.21 Applying Screen Burn-In Reduction Setting [Required], describing how turning power off and on to the Wii Remote Pointer can make the burn-in reduction feature inoperative.
		◆ Rewrote the example of pillarboxing in section 6.23 Support for 16:9 Screen Ratio [Recommended] for easier understanding.
		◆ Updated the content relating to brightness in section 6.24 Avoiding Screen Burn-In Caused by Static Display [Information].
		◆ Added a link in section 7.1.3 Explanatory Message for Wii Remote Strap Use [Required] for applications that support the Wii Zapper. (The link is to 7.4.1.)
		◆ Added a note in section 7.2.4 Disabling Pointer for Applications with Pointer Function [Recommended] regarding the burn-in reduction feature when the `KPADEnableDPD` or `KPADDisableDPD` functions are used.
		◆ Corrected the Japanese version of section 7.2.25 Use of Function WPADSetSpeakerVolume Prohibited [Required].
		◆ Added the new section 7.4 Wii Zapper
		◆ Clarified that the restriction in section 8.2 Limits on Frequent Rewrites [Required] is in regard to updating the FAT. Added comments about temporary editing and replaced the text about the NANDSafe series with text about the `NANDSimpleSafe` series API for auto-saving,
		◆ Revised section 8.11 Notification when the Number of inodes or the Amount of Free Memory is Insufficient [Required] replacing the `OSReturnToMenu` function with the `OSReturnToDataManager` function, which can update the Wii Menu and return to the Data Management Screen. Clarified that a message such as NAND_10 from the Wii message list should be displayed if not moving to the Wii Menu's Data Management Screen.
		◆ Moved the contents of section 8.18 Save Banner File Creation to the subsection 8.18.1 Timing of Creation [Required] and added the section 8.18.2 Check the Save Banner File [Required]
		◆ Deleted mention of a bug workaround in section 8.20 Save Data Comments [Required] because the Wii Menu has been updated.
		◆ Deleted section 8.27 [N. American Version Only] Considerations for Save Banners that Do Not Have Alpha Values Applied because the Wii Menu has been updated so this is no longer necessary.
		◆ Deleted all sections of Chapter 9 Nintendo GameCube Memory Card because the use of the Nintendo GameCube Memory Card has been prohibited and the CARD library will be deleted from the soon-to-be-released Revolution SDK.
		◆ Added the subsection 11.14 Displaying Error Messages for Errors that Occurred While in the HOME Menu [Information] to chapter 11 HOME Menu.
		◆ Deleted section 13.3.1 Banner Icons because starting with Revolution SDK Extensions 2.1, Wii games can also specify DS Single Card Play icons.
The revision history for prior versions was moved to the end of this document.

1 Overview

Wii Programming Guidelines provides programming cautions for the Wii console and peripherals.

1.1 Level of Importance

The following levels indicate the importance of the individual programming guidelines.

- [Required]   Items that must be implemented
- [Recommended]   Items that should be implemented

In addition, related items may be marked as [Information].

1.2 Wii Programming Guidelines Note

The Wii Programming Guidelines are established to reduce problems in the market. However, following these guidelines does not guarantee that all problems will be avoided.

1.3 Do Not Use Wii Development and SDK Files on Other Platforms

Do not use the files included in the various Wii SDKs and development tools on other platforms.

2 All

2.1 Prohibition of Dependency on Device Specifications [Required]

Avoid designing a program with a dependency on the device specifications of the Wii console or its peripherals, since this may result in a program malfunction or crash.

There will be some specification variance between devices. Performance will also decrease as the device ages or a disc becomes damaged.

Specifically, the devices here refer to the disc drive, disc, Wii Remote, Wii console NAND memory, and SD Memory Card. See the corresponding chapters for details.

2.2 Memory and Register Initialization [Required]

The contents of memory are not guaranteed during application program startup. Do not assume a specific initial memory value if memory has not yet been initialized by the application. Similarly, do not use an uninitialized register assuming a specific initial value when using the assembler.

The arena is cleared to zero using OSInit.

2.3 Startup Time Restriction [Recommended]

Display the Strap usage screen within two seconds of switching to the application.

If the startup time is excessively long, the user may mistake this for a system malfunction.

This does not include the 100 or so frames necessary for switching the display mode from either progressive display to interlaced display, or from EURGB60 display to PAL display.

For more information about the Strap usage screen, see section 7.1.3 Explanatory Message for Wii Remote Strap Use [Required].

2.4 ~~Startup Disc Error Handling [Required]~~

(Moved to Chapter 3 Wii Game Discs and Drive.)

2.5 Compliance with Legal Rights Display [Required]

A separate indication of legal rights is required for some of the library tools provided by Nintendo. Use the specified display method when using a library tool or other item requiring a legal rights display.

Not displaying the legal rights information in the specified method may result in licensing issues.

Note that the legal rights display can be shown in various ways, such as during game startup, in the user's manual, and on the packaging. For details, see the instructions for the individual library tools.

2.6 Using the Latest Packages [Recommended]

Always try to use the latest versions of the Revolution SDK and its patches and related tools distributed by Nintendo.

Revisions are made when problems arise.

2.7 Removal of Debug Information [Required]

When you are creating the master data, be sure to set the options so that debug information is not included during compiling and linking.

Also, make sure to use the release versions and not the debug versions of the libraries that you link. When using `make`, set `NDEBUG=TRUE` to disable the OS debug code.

If the debug information is left in the master data, it increases program size in addition to decreasing execution speed. Leaving in debug code output may also result in lock-ups.

2.8 Prohibition of Device Access Without the Use of a Library [Required]

Do not access individual devices directly without going through the appropriate library.

Problems regarding device compatibility and hardware restrictions were taken into account when the various device libraries of the Revolution SDK were created. Unexpected problems could arise if you directly access devices instead of going through the libraries.

2.9 Prohibition of Development Support Libraries in Release Version [Required]

Do not include development support libraries, such as the Host I/O and MIDI adapter libraries, in the release version master data.

Including development support libraries may cause the Wii system to malfunction.

2.10 Guaranteeing Extended Operation of the Application [Required]

Make sure that no fatal bugs such as erroneous processes or lock-ups occur, even if the application is run for a long time.

For example, if continually demonstrating the application overnight in stores, it is conceivable that it will be used continuously for 24 hours or more.

Be careful to avoid lock-ups or other abnormalities due to counter overflows.

2.11 ~~Integrating Banners, Comments, and Icons [Information]~~ [Deleted]

~~Refer to section 8.20 – Save Data Comments [Required] and chapter 12 – Icons and Banners~~

2.12 Application of Wii System Settings [Required]

As applicable, apply the various settings configured for the Wii console to the application program. to allow easier sharing of user-configured values between applications.

For details, see the following guidelines for items in Wii System Settings.

4.1 Applying Wii System Sound Settings [Required]

6.4 Settings and Display when Progressive Display Is Supported [Required]

6.10 Settings and Display for EURGB60 Supported Application [Required]

6.16 Settings and Display for Applications that Support 16:9 Screen Ratio Display [Required]

6.21 Applying Screen Burn-In Reduction Setting [Required]

7.1.3 Explanatory Message for Wii Remote Strap Use [Required]

11.3 Display Language for the HOME Menu [Required]

11.4 Screen Ratio of the HOME Menu [Required]

11.8 Restrictions on Rumble Feature

- 11.8.1 Restrictions Due to the Rumble Setting of the HOME Menu [Required]

- 11.8.2 Prohibition of Changing the Rumble Setting of the HOME Menu [Required]

2.13 Prohibition of Health and Safety Warning Display by Applications [Required]

The Wii displays a "Health and Safety Precautions" warning message at startup of the Wii console. Nintendo considers this warning screen a blanket warning, so do not include a health and safety warning screen on the application side. There is no problem with displaying a warning screen for the rating of the game's content.

~~2.14 Display at Application Startup [Information]~~ [Deleted]

~~There is no need for the application to display any warning at startup other than the "Health and Safety Precautions" warning message.~~

2.15 [N. American Version Only] Displaying the ESRB Online Rating Notice for Applications Using Network Features [Required]

If applications released in the U.S. market that are subject to ESRB rating have unrestricted communication between users during a network connection and/or allows users to send and receive elements (user-generated data, etc.) not rated by ESRB, it will be noted as a requirement on the "ESRB Rating Certificate" to display the ESRB Online Rating Notice. If the display of the notice is required by ESRB, see the ESRB website (https://www.esrb.org/) to confirm the detailed ESRB requirements (see ESRB ARC Manual) and display the notice.

2.16 Program Distribution Method [Required]

Do not run programs other than the following:

- Programs included on Wii Game Discs produced by Nintendo

- Programs distributed by the program distribution system provided by Nintendo.

Do not write or read in programs from the application using Wii Remote memory, an SD Card, a network, a USB port, Bluetooth, or other kind of interface.

Also even if data is simply being read using these interfaces, always perform buffer overflow checks and use data signature, authentication and encryption to ensure programs will not be executed with buffer overflow due to modification of data

2.17 Parental Controls [Information]

The Wii system can restrict the startup of Wii games based on a Parental Control feature. This feature can be used when a parent or guardian does not want the system to allow access to games that are inappropriate for members of their household. The parent or guardian can set the restriction level based on a game's rating. The parent or guardian will set a PIN for temporarily releasing the Parental Controls.

Games for the Wii system must be rated by the rating agency for each market in which the game is released before submitting a master. When creating master data for submission, be sure to enter the rating using the Master Editor program.

Table 2-1 lists the rating agencies for each market.

Table 2-1   Rating Agencies by Market

Market	Rating Agency
Japan	CERO
North America	ESRB
Europe	PEGI, USK, BBFC (varies by country)
Australia	OFLC

The Wii console restricts the startup of games according to the restriction level set by the parent or guardian based on the rating stored in the game. Restricted games can be started only if the correct PIN is entered when the user is prompted for it immediately before startup. Note that applications do not need to perform any special processing for the restrictions based on their rating.

For details, see the *Revolution Master Editor* manual.

2.18 [North American, European, and Australian Versions] License Display of Licensee Titles [Required]

Licensee titles must display the text "Licensed by Nintendo" for one second or more in one of the following places:

- Banner for the Disc Channel of the Wii console

- Application startup sequence (after the Strap Usage Screen if that screen is required)

The font used for the text "Licensed by Nintendo" should be of a size and type that can be read by the user. There is no requirement for the display location.

This message can be displayed with other rights information.

If "Licensed by Nintendo" will be displayed in the banner, refer to section 12.7 Display of Trademarked or Copyrighted Names in the Banner [Information]

To read about the Strap Usage Screen, see 7.1.3 Explanatory Message for Wii Remote Strap Use [Required].

2.19 Using Firmware of Specified Versions [Required]

For applications developed with the Revolution SDK 3.1 or later environments, use the "extension firmware" (firmware for titles with support for network functionality) when using Revolution SDK Extensions (REX) or Revolution DWC (DWC), and use the "standard firmware" when they are not used.

To determine the specific version of firmware to use, see the *Revolution SDK README*.

2.20 Time Restriction when Screen Is in Static State [Required]

Do not allow the screen to be in a static state for more than 10 seconds.

The reason for this restriction is that users may become uneasy if the screen remains in an unchanged state for a long period, and may suspect that the Wii console is damaged.

For example, the screen will remain black for a long time if a large DOL file has been specified by the `OSExec*` function, so be aware of the size of the DOL file.

Note that this restriction does not apply if the user can confirm that the screen or the audio will change with controller operations.

Contact support@noa.com if effects in the game require the screen to remain in a static state for longer than 10 seconds, even if the user can confirm that the screen or the audio will change with controller operations.

2.21 Terminology and Image Specifications

2.21.1 Standardized Terminology [Required]

Adhere to the latest version of the Wii Terminology document for the approved terminology of the Wii console, various parts of the Wii console, terms relating to operation, names of peripheral devices, and all other names.

2.21.2 Restrictions on Images Used Instead of Names [Recommended]

For reasons of space and design you might want to use images instead of names, but try to use images that are as close to actual shapes as possible. A little variation in color or design is acceptable if it will not be confusing to users.

2.22 Using the Time from the Wii System Clock

The Wii system clock time can vary greatly if the battery runs out or is changed. If the battery runs out, the time from this clock is undefined.

Applications that use the time from the Wii system clock should support the following items:

2.22.1 Procedure when Time of Wii System Clock Differs Greatly from Real Time [Required]

Do not design your application in such a way that the game's progress is hindered when the time on the Wii system clock differs greatly from the actual time.

2.22.2 Prompting a User to Check the Current Time [Recommended]

We recommend that you display the time of the Wii system clock and prompt the user to confirm the actual time in the following situations:

- At the time of the first startup

- When the current time is significantly off from the previous startup

Users probably take little notice of a deviation in the time, but by prompting them to confirm the time you can get their attention when the Wii system clock develops a greatly different time, such as when the battery suddenly dies.

3 Wii Game Discs and Drive

3.1 Prohibition on Drive Specification Dependency [Required]

Ensure that programs do not run uncontrollably or freeze due to a dependency on drive access speed.

Drive access speed differs depending on the status of the disc and/or drive. Furthermore, specifications regarding drive access speed, cache size, and cache operations may change in the future.

Be sure to thoroughly debug programs so that they do not freeze even when drive access speed suddenly drops, particularly in cases where data is used sequentially, as in movie scenes.

3.2 Wii Game Disc Error Handling [Required]

For Wii Game Disc error handling, adhere to the content in *Wii Optical Disc Drive Guidelines*, unless there is a special reason not to do so.

Having each application handle Wii Game Disc errors differently confuses users.

If there is a special reason to change error handling, contact support@noa.com early in development.

3.3 Prohibition of Sustained Continuous Non-Sequential Access [Required]

If there has been no user input for more than five minutes (or 10-15 minutes, based on the screen burn-in reduction feature setting), continuous non-sequential disc access should end within one hour. Once user input is received, resume normal operations.

Non-sequential access is defined as seeking to access data spaced more than 200 MB apart on the disc. Non-sequential access resumed within five seconds for a long period of time can shorten the lifespan of the disc drive. To avoid unnecessary aging of the disc drive while the user is not operating the application, do not conduct this kind of non-sequential access for any longer than one continuous hour.

For example, when a movie is playing for a long time, position the files that will be accessed nearby and, if non-sequential access will be carried out, limit the number of loops.

For information on the wait time set for the screen burn-in reduction feature, see the *Wii Video Interface Library (VI)* manual and the *Video Interface Library* section of the *Revolution Function Reference Manual*. If you are going to reconfigure the wait time for the screen burn-in reduction feature, see section 6.22 Changing Screen Burn-In Reduction Wait Time [Recommended].

3.4 [Multiple Disc Games Only] Independent Startup of All Discs [Required]

Even if startup of the application from a particular Wii Game Disc is not expected, always make sure that some kind of program is executed when the disc is started independently, even if it is just a warning message. For example, if the inserted Wii Game Disc is one of a set of two Wii Game Discs and the inserted disc is one that cannot be played as a standalone disc, prepare a program that displays a message similar to DISC_06 in the *Wii Message List*.

Not doing so may cause the user to think that there is malfunction in either the Wii console or the Wii Game Disc when a Wii Game Disc is not designed for independent startup.

3.5 ~~Normal Operation after the Disc Drive Motor Is Shut Off From a Period of Inactivity [Required]~~ [Deleted]

(Deleted because shutting off the disc drive motor does not affect the development environment currently provided.)

3.6 ~~Display for Disc Drive Motor Shut Off From a Period of Inactivity [Recommended]~~ [Deleted]

(Deleted because shutting off the disc drive motor does not affect the development environment currently provided.)

3.7 Data Layout for Double Layer Disc

Applications that support double-layer discs must support the following items:

3.7.1 Avoid Disc Access Spanning Layers [Required]

Place a 16MB dummy file (a file that the application does not use) starting from address 0xE7000000. The actual starting location of the second layer on the disc is 0x0E78A6C00, but for simplicity this has been rounded up. By locating this 16 MB dummy file here, you do not need to worry about the possibility of the Read at the layer border having different conditions from the Read at another address.

The following is the specific sample code:

```
[LbaOrigin]
0xE7000000="border.dat",+
```

3.7.2 Avoid Reading Data to Alternate Layers [Recommended]

Avoid data layouts that require alternating reading between the first and second layer. To minimize reading from different layers, extra steps might be necessary, such as placing the same data on both layers.

Switching between the first and second layer adds extra latency.

3.8 Startup Disc Error Handling [Required]

Always include disc error handling routines (including error display) in programs read by the Wii Menu.

If there is no error handling routine (error display) in sections read by the Wii Menu, errors that occur before error handling routines are read will not be processed.

For details on error handling, see the *Wii Optical Disc Drive Guidelines*.

3.9 Resetting and Shutting Down in Response to Fatal Errors [Required]

Disable controller operations, RESET, and the Power Button while a fatal error message is being displayed. Fatal errors indicate a state where it may not be safe to even reset the Wii console, so give priority to displaying the error message without resetting or shutting down.

For more information about fatal error messages, see section 3.11 Message Display in Response to Fatal Errors [Required].

3.10 Resetting and Shutting Down in Response to Disc Errors Other than Fatal Errors [Required]

In the cases of "No Disc," "Wrong Disc," or "Retry" errors, when a reset operation occurs, perform reset processing and return to the Wii Menu. If the Power Button on either the Wii console or the Wii Remote is pressed, perform shutdown of the system immediately.

For information about the optical disc drive library displaying fatal error messages automatically, see Chapter 10 Reset and Shutdown.

3.11 Message Display in Response to Fatal Errors [Required]

If you call the `DVDSetAutoFatalMessaging(FALSE)` function at the start of the application to disable the feature that automatically displays fatal error messages, display `DISC_04` in the Wii Message List in response to fatal errors.

As long as this feature is not disabled by calling the aforementioned function, the optical disc drive library will display fatal error messages automatically. The automatically displayed message will reflect `DISC_04` from the latest version of the Wii Message List at the time the Revolution SDK being used was released. Consequently, the message may be different from the latest Wii Message List depending on which Revolution SDK version and language is being used, but this is not considered to be a problem.

4 Sound

4.1 Applying Wii System Sound Settings [Required]

If there are no menus in the application where the user can select sound settings, apply monaural, stereo, or surround from the Wii system sound settings during startup.

Also, if there is a menu in the application where the user can select sound settings, apply the sound setting of the Wii system as an initial value during the initial startup of the application.

In either case, if the sound setting of the Wii system is either monaural or surround, and the application does not support that setting, output in stereo (support for stereo output is required).

4.2 ~~Sound Setting on the Wii Console [Required]~~ [Deleted]

(Deleted because it was restructured as section 4.1 Applying Wii System Sound Settings [Required].)

4.3 ~~Sound Output for Applications Supporting Multiple Sound Settings [Required]~~ [Deleted]

(Deleted because it was restructured as section 4.1 Applying Wii System Sound Settings [Required].)

4.4 Do Not Modify the Audio Playback Frequency Within an Application [Required]

Although the Wii audio system supports playback at a frequency of either 32 kHz or 48 kHz, once one of these frequencies has been set for an application at startup, be sure not to change the frequency setting again. The ability to change the playback frequency is limited due to hardware specifications.

5 Photosensitivity

5.1 About Photosensitivity and These Guidelines [Information]

These guidelines are intended to be used in the development of video games for the Wii platform. Unlike films and television programs, which produce only one sequence of images each time they are played, one video game can produce an infinite sequence of images. This is because video games are interactive, so that each time a game is played, a different sequence of pictures and images will be displayed, depending on the choices and inputs made by the game's player or—in the case of multi-player games—players. In addition, the luminance of images displayed in three-dimensional games are not simply those of the video game artist's original image but are the result of the game's programming processes, which render the image in a three-dimensional form in a three-dimensional space, with variations of light, shadow, distance, orientation and player perspective. These variables also are affected by choices made by the individual player.

Because of these infinite variations that are possible within a single game, it may be possible with many games that certain player inputs will cause screen imagery that will exceed the suggested limits described below. Developers should try to design games that comply with the limits when the games are played with normal gaming strategies and inputs, with the recognition that it may still be possible for player inputs to cause sequences of images that may exceed the suggested limits, particularly if the gameplay is idiosyncratic or counterintuitive. It must also be remembered that compliance with these guidelines or with any other guidelines that have been or will be developed may reduce the incidence of photosensitive seizures, but they will not eliminate them or eliminate seizures that occur during video gameplay from causes other than the visual content of the games.

These guidelines attempt to take what medical science has learned about the images that can trigger photosensitive seizures in the universe of susceptible individuals and, in a few paragraphs, apply it to the infinite variety of imagery produced by modern video game technology. Medical research in this area is still developing, and the particular susceptibilities of photosensitive persons vary widely from individual to individual. As the developers of other guidelines have recognized, it is impossible to craft guidelines that will eliminate all risk of seizures, and the measures taken should be proportionate to the risks involved and should not stifle developers' creativity, imagination or freedom of expression. It may be possible that a game, even though complying with the guidelines, may produce a problematic sequence. Alternatively, a sequence out of compliance with the guidelines may not be problematic in its context. It is therefore recommended that all games, before final release, be reviewed by one or more persons knowledgeable about photosensitivity who can check for potentially problematic sequences. It is also recommended that such persons review decisions to deviate from the guidelines when that may be desirable for the artistic or creative imperatives of a game.

These guidelines use the following lighting technology terms:

- **Luminance** is a quantifiable measurement of the observed brightness of an object—in this case, of a video screen.

- **Nits** is a shorthand name for candelas per square meter, the metric system's measurement unit for luminance. (A candela is a measure of the candle power or angular density of light from a source).

- A **photometer** is a device that measures the luminance of an object. A photometer with CIE characteristics is calibrated to match the response to various color spectra of the average human eye.

- The **RGB value** of a color in a video display is a three-number representation of the intensities of, respectively, the red, green and blue elements of the display that combine to form the color. Each value is a number from 0 to 255. Consequently, an RGB value of (0,0,0) is black; and RGB value of (255,255,255) is white; and an RGB value of (255,0,0) is pure red.

A video with sample footage has been prepared to illustrate and supplement the guidelines. When a portion of the guidelines is illustrated by the video, the guidelines includes a reference to the relevant section of the video. The video provides supplemental illustrations and is not an essential part of the guidelines, which can be used without the video.

5.2 Restrictions on Flashing Images and Lights [Recommended]

Do not use a sequence of images that

(a)   flashes so that the change in luminance of the flash exceeds 20 nits (candelas/meter^2), and

(b)   occupies more than one quarter of the screen, and

(c)   has more than three flashes occurring in any one second period.

The sample video contains examples of luminance changes of different magnitudes in sections 1 (1), 1 (2), and 1 (3).

A flash is a pair of opposing changes in luminance—that is, an increase in luminance followed by a decrease or a decrease followed by an increase. If the luminance measurements of successive flashes over time are plotted using x and y coordinates (x=time; y=luminance), the shape of the resulting plot will appear in profile as alternating peaks (frames of localized maximum brightness) and valleys (frames of localized minimum brightness). Flashes should be evaluated for the change in luminance between adjacent peaks and valleys. In games for NTSC monitors, no more than three of these peaks (or, alternatively, no more than three valleys) should occur in any thirty consecutive frames. In games for PAL monitors, no more than three peaks (or, alternatively, no more than three valleys) should occur in any twenty-five consecutive frames.

Screen luminance can be measured or calculated as described in section "5.6 Screen Brightness Calculations [Information]".

5.3 Restrictions on Flashing Saturated Red Colors [Recommended]

Do not use a sequence of images in which

(a)   the images produce flashes (regardless of the change in luminance of the flashes), and

(b)   one of the images contains saturated red, and

(c)   the saturated red occupies more than one eighth of the screen, and

(d)   more than three flashes occur in any one second period.

The sample video contains examples of red flashing in sections 2 (1), 2 (2), and 2 (3).

Saturated red is a color whose RGB value for red is greater than 85% of the sum of the color's RGB values.

5.4 Restrictions on Image Reversals [Recommended]

If the luminance of the elements of an image that occupies more than one quarter of the screen are switched or interchanged (for example, switching between the negative and positive of an image or black and white images in which the black turns white while the white turns black, as in the samples below), the changes in luminance should not exceed 20 nits or occur at a rate faster than that allowed for flashing in Section 1.

The sample video contains examples of images with switching luminance in sections 3 (1) and 3 (2).

5.5 Restrictions on Regular Patterns [Recommended]

Do not use an image that

(a)   consists of striped patterns composed of parallel lines or dots or other regular elements with distinct edges, such as the samples below, and

(b)   has "high contrast" between the bright and dark elements of the pattern, as defined below, and

(c)   occupies more than one quarter of the screen, and

(d)   has more than five light–dark pairs of stripes in any orientation.

An image has "high contrast" when

(i)   the luminance of the brighter element of the pattern is 30 nits or more, and its contrast is greater than 40% (contrast is (L1-L2)/(L1+L2), where L1 is the luminance of the brighter element of the pattern and L2 is the luminance of the darker), or

(ii)   the luminance of the brighter element of the pattern is less than 30 nits and the difference in luminance between the brighter and darker elements (L1-L2) is 17 nits or more.

The sample video contains examples of patterns in sections 4 (1), 4 (2), 4 (3), and 4 (4).

The stripes may be parallel or radial, curved or straight, black and white, or a combination of colors. Avoid especially stripes that oscillate or flash and moving stripes that change direction. Do not switch the luminance of the lighter and darker stripes (so that the dark become light and vice versa). Striped patterns that flow smoothly across, into, or out of the screen in one direction may be used. Checkerboard patterns and plaids are acceptable.
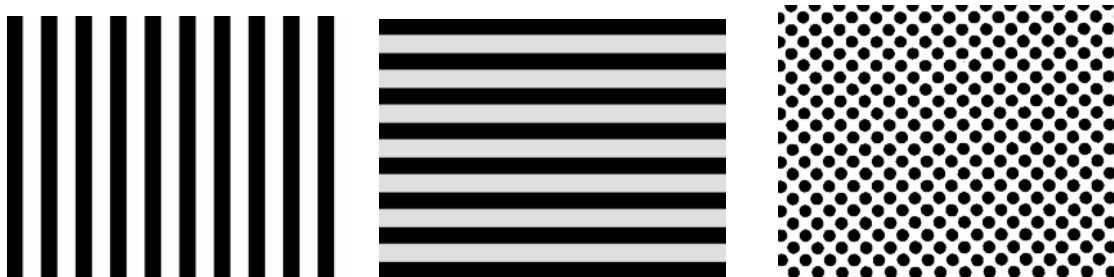
5.6 Screen Brightness Calculations [Information]

Three methods can be used to measure luminance for games displayed on a cathode ray tube (CRT) TV screen.

1. First, screen luminance can be measured directly from a monitor with a hand-held spot photometer with a CIE characteristic designed for making measurements from a television screen. For consistency, the display brightness and contrast should first be set up using PLUGE (Rec. ITU-R BT.814) with peak white corresponding to a screen illumination of 200 nits. This corresponds to the display conditions of the "home viewing environment" described in Recommendation ITU-R BT.500.

2. Second, the luminance (monochrome) voltage can be used to determine the emitted light output of a television display that matches the "home viewing environment" described in the preceding paragraph. The following graph and table show the typical relationship between voltage and screen luminance for such a display in the NTSC format.

Screen Light Output in Nits versus Luminance Voltage
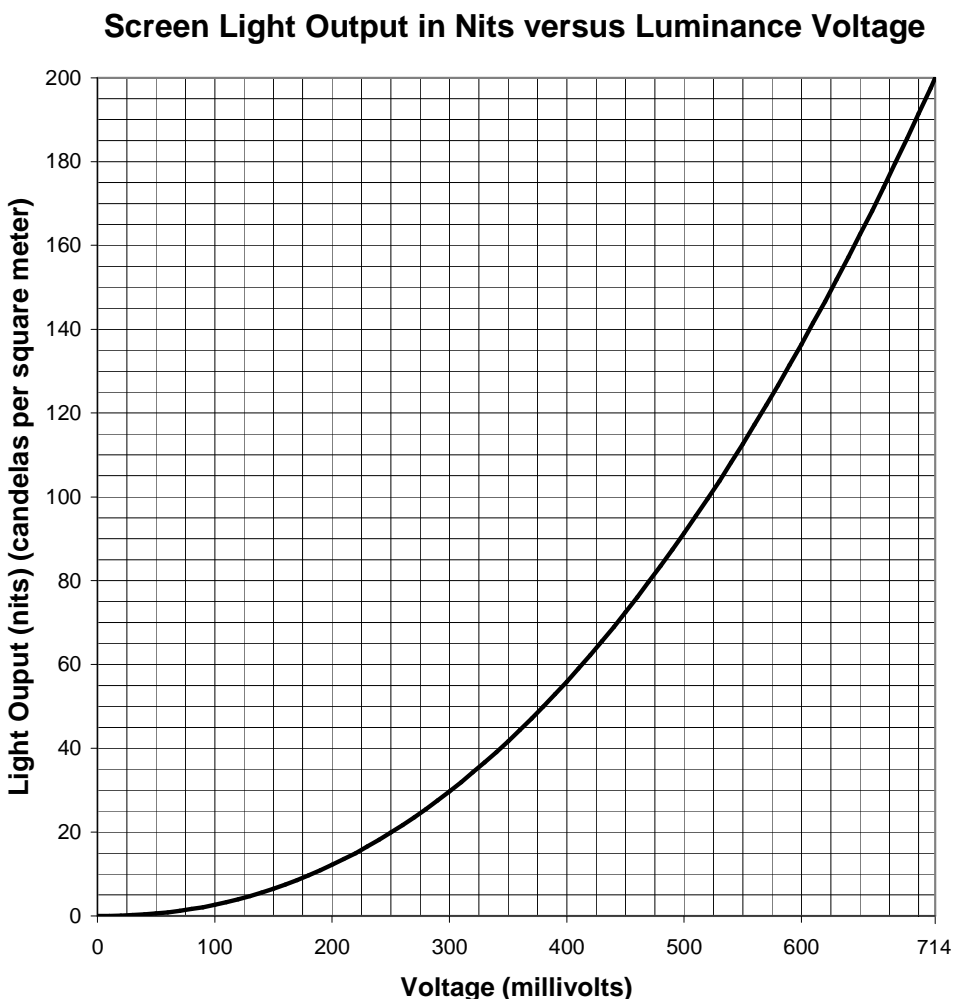
Voltage (mV)	Light (cd/m²)
0	0.00
25	0.13
50	0.58
75	1.41
100	2.65
125	4.33
150	6.46
175	9.07
200	12.17
225	15.76
250	19.88
275	24.51
300	29.69
325	35.40
350	41.67
375	48.50
400	55.90
425	63.88
450	72.44
475	81.59
500	91.33
525	101.68
550	112.64
575	124.21
600	136.40
625	149.22
650	162.67
675	176.75
700	191.47
714	200.00

The relationship between voltage and luminance is determined by the following formula:

$$L = \left(\frac{mV}{714}\right)^{2.2} \times 200$$

In this formula, **L** is luminance in nits or candelas per square meter and **mV** is luminance voltage in millivolts.

3. Third, luma (Y), which the Wii console converts and outputs, can be calculated from the RGB values (from 0 to 255 each) of the Wii software side, by using the following formula:

$$Y = 0.257 \times R + 0.504 \times G + 0.098 \times B + 16$$

Y is a value between 16 and 235. When NTSC composite video or S terminal video outputs white 100%, the analog output voltage is 714 mV (except the synchronized level). These relationships result in the following formula for converting RGB into screen luminance in nits on an NTSC display:

$$L = \left[\frac{0.838 \times R + 1.643 \times G + 0.319 \times B}{714} \right]^{2.2} \times 200$$

Therefore, the luma, analog voltage, and luminance of each color is as follows, when a standard color bar is displayed:

Display Color (R/G/B)	Y (Digital Value)	Analog Voltage (mV)	Luminance (nits)
White (255/255/255)	235	714	200
Yellow (255/255/0)	210	633	153
Light Blue (0/255/255)	169	501	92
Green (0/255/0)	144	419	62
Purple (255/0/255)	106	295	29
Red (255/0/0)	81	214	14
Blue (0/0/255)	40	81	2
Black (0/0/0)	16	0	0

The preceding calculations assume that the programmer has selected a gamma (γ) of 1.0, as recommended in the section "Gamma Correction" in the Revolution Graphics Library, contained in your Revolution SDK. If a different gamma is selected, it must be accounted for in the formula:

$$L = \left[\frac{213.714 \times \left(\frac{R}{255} \right)^{\frac{1}{\gamma}} + 418.169 \times \left(\frac{G}{255} \right)^{\frac{1}{\gamma}} + 81.498 \times \left(\frac{B}{255} \right)^{\frac{1}{\gamma}}}{714} \right]^{2.2} \times 200$$

6 Video

6.1 Display Within the Safe Frame [Recommended]

Decide on the display position based on the importance of the information to be displayed.

The size of the safe frame in Wii depends on the specific TV set. However, in general, with a 4:3 aspect ratio, the safe frame width will be approximately 84.4% of the maximum valid horizontal display width in the horizontal direction and approximately 80.8% of the maximum valid vertical display height regardless of the display format. With a 16:9 aspect ratio, the safe frame horizontal width and vertical height will both be approximately 87% of the maximum valid display frame.

For details on safe frames, see the *Wii Video Interface Library (VI)* manual.

~~6.2 Video Settings [Information]~~ [Deleted]

(Deleted because this content overlapped with other paragraphs in Chapter 6 Video.)

6.3 Support for Progressive Display [Recommended]

We recommend adding application support for progressive display output.

However, this recommendation need not be followed when it is not feasible because field rendering is used or for other reasons.

6.4 Settings and Display when Progressive Display Is Supported [Required]

If the application supports progressive display, support the display format selected on the Wii console's TV Resolution setting screen to present either an interlaced display or progressive display.

6.5 Procedure for Switching from Progressive Display to Interlaced Display [Recommended]

If the scan mode is progressive display when your application starts, but your application does not support progressive display, use the following procedure to switch the screen to interlaced display.

[Black screen] → [Switch from progressive to interlaced] → [Black screen] → [Wait for around 100 frames] → [Strap Usage Screen]

Only do this switch once. During the switch between interlaced and progressive display, the screen will be distorted and require a few dozen frames before returning to normal.

For details on display switching, see "Recommendations for Progressive Display" in the *Wii Video Interface Library (VI)* manual.

To read about the Strap Usage Screen, see 7.1.3 Explanatory Message for Wii Remote Strap Use [Required].

6.6 ~~Addition of Interlaced/Progressive Switch Operation~~ [Deleted]

(This has been deleted because there is no operation switching display mode from interlaced to progressive.)

6.7 ~~Reset Operation During Progressive Output~~ [Deleted]

(This has been deleted because the information duplicates the content in 6.5 Procedure for Switching from Progressive Display to Interlaced Display [Recommended])

6.8 Televisions that Support Progressive Output [Recommended]

Avoid displaying important messages in small fonts, natural images, or per frame flash effect in non-interlaced (double-strike mode) display.

Some televisions that support progressive output may perform scan line interpolation. You may not be able to achieve the desired screen display on such a television.

6.9 PAL 50 Support [Required]

Always support PAL 50 in the European and Australian versions of the application.

6.10 Settings and Display for EURGB60 Supported Application [Required]

If the application supports EURGB60, display in either PAL 50 or EURGB 60, based on the mode configured on the Wii console.

6.11 Switching from EURGB60 to PAL [Recommended]

If the TV mode is EU RGB 60 when your application starts but your application does not support EU RGB 60, use the following procedure to switch the screen to PAL.

[Black screen] → [Switch from EU RGB 60 to PAL] → [Black screen] → [Wait for around 100 frames] → [Strap Usage Screen]

Only do this switch once.

During the switch between PAL and the EURGB60 display, the screen will be distorted and require a few dozen frames before returning to normal.

For details on display switching, see "EURGB60 Mode" in the *Wii Video Interface Library (VI)* manual.

For details on the Strap usage screen, see section 7.1.3 Explanatory Message for Wii Remote Strap Use [Required].

6.12 ~~Addition of PAL/EURGB60 Switch Operation~~ [Deleted]

(This has been deleted because there is no operation switching from PAL to EU RGB 60.)

6.13 ~~Reset Operation During EURGB60 Output~~ [Deleted]

(This has been deleted because the information duplicates the content in 6.11 Switching from EURGB60 to PAL [Recommended]

6.14 ~~Maintaining Appropriate Video Output~~ [Deleted]

(This has been deleted because this is checked by the Video Interface library, so the application does not need to support it.)

6.15 Display Method for 16:9 Screen Ratio [Recommended]

When supporting a 16:9 screen ratio, we recommend implementing the anamorphic widescreen technique.

The anamorphic widescreen technique horizontally compresses 16:9 aspect ratio video so that it ends up with a 4:3 aspect ratio. The technique produces widescreen video with higher quality and higher resolution than the letterbox method.

Anamorphic widescreen video can be restored to the original 16:9 video by having the television stretch out the display uniformly in the horizontal direction. This type of television display mode is generally referred to as "full," and most widescreen televisions support display in full mode.

Moreover, Wii prompts users to set the television to full mode when the screen mode has been set to 16:9 in the Wii System Settings.

For details, see the *Wii Video Interface Library (VI)* manual.

6.16 Settings and Display for Applications that Support 16:9 Screen Ratio Display [Required]

If an application supports the 16:9 screen ratio, be sure that the display is done in either the 4:3 or 16:9 aspect ratio, depending on the aspect ratio set for the Wii console.

Although it is acceptable for an application to have a separate aspect ratio setting, the Wii console settings should be referenced when the application is first launched and reflected in those separate settings.

Applications not supporting the 16:9 screen ratio may ignore the Wii console setting.

6.17 Trap Filter [Recommended]

When there is a large contrast in brightness, such as with a white font on a black screen or a black font on a white screen, and a small font size is used, please enable the trap filter.

If the trap filter is disabled when displaying such an image on some CRT television sets with high contrast, the effect of cross color may become pronounced and distort the outlines of characters.

6.18 Support for 4:3 Aspect Ratio [Required]

Applications must always offer support for the 4:3 aspect ratio. Do not create applications on the assumption that they will only be displayed in the 16:9 aspect ratio.

For details, see the *Wii Video Interface Library (VI)* manual.

If your application is based on the mode with 16:9 aspect ratio, rather than preparing a separate screen with 4:3 aspect ratio you could instead create a letterboxed screen with bands at the top and bottom and use it as the 4:3 aspect ratio mode.

For more information on displaying static images, see section 6.24 Avoiding Screen Burn-In Caused by Static Display [Information]

6.19 Checking the On-Screen Display During Application Startup [Required]

Regardless of the display mode used for the Wii menu, you must verify that an application can be displayed normally. Although applications are not required to support all display modes (progressive, EURGB60, and 16:9), the Wii menu does include all display modes. For example, applications that do not support progressive display must be displayed normally even if the Wii menu is being displayed in progressive mode.

For more information, see the *Wii Video Interface Library (VI)* manual.

6.20 ~~Check Before Switching to Progressive Display~~ [Deleted]

(This has been deleted because it can be checked with the `VIGetScanMode` function.)

6.21 Applying Screen Burn-In Reduction Setting [Required]

When the screen burn-in reduction setting on the Wii console is ON, do not disable the screen burn-in reduction feature with the application.

In situations where the power to the Wii Remote Pointer will be turned off and on repeatedly, make sure this does not have the effect of repeatedly resetting the screen burn-in reduction feature, since this in effect renders the feature inoperative. (See 7.2.4 Disabling Pointer for Applications with Pointer Function [Recommended] to read about turning the power to the Pointer on and off.) If it is necessary to disable the screen burn-in reduction feature due to application specifications regardless of the console settings, first determine whether to disable the feature after meeting the following two conditions and giving due consideration to whether there is a possibility of screen burn-in occurring.

- Prepare video appropriate for televisions of 4:3 and 16:9 aspect ratios.
  By supporting televisions of both 4:3 and 16:9 aspect ratios, you can supply the appropriate video output regardless of the user's television aspect ratio.

- The display of all pixels output from the Wii console changes.
  Even if characters are moving, if the camera is fixed and there is a fixed display, such as TIME or

SCORE, then it cannot be said that the display for all the pixels changes. So, do not disable screen burn-in reduction.

For details, see the *Wii Video Interface Library (VI)* manual and function reference.

6.22 Changing Screen Burn-In Reduction Wait Time [Recommended]

Unless required otherwise, do not change the wait time before the screen burn-in reduction from the default of 5 minutes. In addition, if the value is to be changed, do not set a value longer than required.

6.22.1 ~~Restrictions for Non-Activation of Screen Burn-in Reduction [Required]~~

(Deleted because it was restructured as section 6.21 Applying Screen Burn-In Reduction Setting [Required] and 6.22 Changing Screen Burn-In Reduction Wait Time [Recommended].)

6.23 Support for 16:9 Screen Ratio [Recommended]

Support for 16:9 screen ratio is recommended.

This is because 16:9 widescreen televisions are expected to become predominant in the future.

For details, see the *Wii Video Interface Library (VI)* manual.

If your application is based on the mode with 4:3 aspect ratio, rather than preparing a separate screen with 16:9 aspect ratio you could instead create a pillarboxed screen with bands on the sides and use it as the 16:9 aspect ratio mode.

See 6.24 Avoiding Screen Burn-In Caused by Static Display [Information] for more information on displaying static images.

6.24 Avoiding Screen Burn-In Caused by Static Display [Information]

Although the Wii console has a feature to decrease display brightness in order to reduce screen burn-in, be aware that other factors (such as those listed below) may cause screen burn-in, depending on the television's display method.

- Only the area of the 4:3 aspect ratio is drawn for a 16:9 display, and the margins to the left and right are filled in with static images.

- The margins above and below the drawing area, such as during playback of movies, are filled in with static images.

- Text such as times and scores that do not change for long periods and are displayed continuously.

In general, when static images or non-moving text   are displayed brightly (that is, displayed using colors that are close to the primary colors), screen burn-in will be particularly noticeable for text, its surrounding area, and for boundary lines between static images and animated areas.

If it is necessary to generate a static display, such as in the examples listed above, you may want to consider applying the following methods in order to avoid screen burn-in.

- Confine to 15 minutes situations wherein the same [content] is displayed continuously.

- Use a display that has low overall brightness.

- Change the brightness of the display

- Display margins with the same approximate brightness as the drawing area.

- When changing screens, use the same color or pattern as the margin uses for the entire screen.

7 Controllers

7.1 All Controllers

7.1.1 Display when Extension Controller is Necessary [Required]

Display a message regarding any required extension controller at application startup or before the extension controller is required by the application. This message should be similar to CONT_01 or CONT_02 in the *Wii Message List*.

The purpose of this requirement is to inform the user of the need for a particular extension controller for gameplay.

However, if an existing connection of the extension controller required by the application has been confirmed, there is no need to display the message.

7.1.2 Avoiding Game Design Forcing Extended Continuous Operation or Strenuous Activity [Information]

To prevent the user from experiencing physical injuries such as RSI, avoid game designs that force the following types of controller operations for extended periods of time.

- Strenuous actions, such as swinging around

- Repetitive motion in a single direction

- Keeping the controller frozen in a stationary position

In particular, make sure the game design does not require users to compete using strenuous actions or to repeat strenuous actions for an extended period of time to progress in the game.

We recommend inserting an appropriate number of scenes where no controller operations are necessary.

Also, during development, use multiple game testers to evaluate whether excessive stress is placed on the body.

7.1.3   Explanatory Message for Wii Remote Strap Use [Required]

When starting up an application using the Pointer, Motion Sensor, or both, display a message prompting the user to check that the Wii Remote strap is attached. The message should also indicate that when using a controller connected to the Wii Remote, including the Nunchuk, put the wrist strap through the connector hook.

Furthermore, be sure to use the image and text included in the "Wii Strap Reminder package" to prompt strap attachment. Unless you have obtained Nintendo's prior permission, this image and text must be used by all software. In addition to the image and text for prompting use of the strap, the prompt screen also includes an image and text for prompting use of the safety hook when using the Nunchuk. See the "Wii Strap Reminder package" for the display sequence of these texts and images.

Select the text and images to be displayed in the strap attachment prompt screen from English, French, and Spanish for North American market software; and English, German, French, Spanish, Italian, and Dutch for European market software, based on the language setting of the Wii console. Note that this language selection is separate from the language supported by the software application.

There may be games where this requirement is not necessary, such as where the controller use is limited to horizontal use of the Wii Remote and you are holding it with both hands. Contact support@noa.com if you believe your game design does not require the wrist strap reminder screen.

For applications that support the Wii Zapper, see 7.4.1 Displaying the How to Hold the Wii Zapper Screen [Required].

7.1.4   Handling +Control Pad Input [Required]

Make sure that simultaneously pressing the up and down or left and right directional buttons on the +Control Pad of the Wii Remote, Classic Controller, and Nintendo GameCube Controller does not cause program malfunctions or other problems.

Up and down input or left and right input can be provided at the same time depending on the usage environment.

Some of the ways to handle simultaneous pressing of the up and down or left and right buttons are as follows.

- Attach priority levels to up/down and left/right directions.

- Treat the input as null (no input).

Although prohibition processing takes place in the WPAD library for the +Control Pad on both the Wii Remote and Classic Controller, other measures are required to handle OR'ed input from both.

7.1.5 Operations on the Unused Buttons [Required]

Make sure there is no program malfunction when an unused button is pressed. This also applies to cases where second or third controllers are used or where a non-supported extension controller is used in an application that uses only the first controller.

There is a chance for a button not used in the application to be input. Unexpected inputs may also come from peripheral devices.

Be especially careful when a button not used in the application can perform actions, such as the startup of debug mode.

7.1.6 Lower Limit Value of the Control Stick and Button Analog Input Value [Recommended]

To prevent the application from acknowledging input when Control Sticks or buttons are not being used, give an extra allowance to the lower limit value.

When a user is not touching the Control Sticks or any buttons, the coordinates given by the Control Sticks or the buttons will not necessarily be those of the origin point. The offset given here will vary for each controller, and will further show some variance each time the controller is used.

See the entries for WPADClampStick and the various clamp functions in the PAD library.

7.1.7 Upper Limit Value of the Control Stick and Button Analog Input Value [Required]

The upper limit of the analog input value from Control Sticks and buttons will vary for each controller. Do not create an operation that requires a value that is impossible to input.

Keep the Control Stick and button analog input value within the range set by Nintendo.

The resolution of individual Control Sticks and buttons will show some variations between controllers. The upper limit value for the same controller may also decrease over time due to age-related wear.

In addition, each input value will have limits based on the shape of the controller frame. Make sure that operation is not affected when the controller frame shape changes to expand the movement range, as this will change the maximum input value.

See the entries for WPADClampStick and the various clamp functions in the PAD library.

7.1.8 Analog Input Value Range of Control Stick and Button [Required]

Verify proper operation with all Control Stick input values from +127 to -128 in the up/down and the left/right direction and button analog input values from 0 to 255.

The above range of values may be input when the origin is shifted.

See the entries for WPADClampStick and the various clamp functions in the *Nintendo GameCube Controller Library (PAD).*

7.1.9   Regarding Applications that Support the Rumble Feature [Required]

In applications that require the Rumble Feature for the player to progress, make sure to have an alternative presentation using images or sounds when the Rumble Feature is turned off so that the application can progress without problems.

Not doing so will cause problems when the user has turned off the Rumble Feature or is using the WaveBird controller, which has no Rumble Feature.

We consider an application that requires the Rumble Feature to be one that cannot be finished without using the Rumble Feature.

The enabled state of the Nintendo GameCube Controller Rumble Feature can be checked with the return value of the `PADRead` function.

7.1.10 Avoiding Continuous Operation of the Rumble Motor [Recommended]

If possible, avoid continuous operation of the Rumble Motor.

A Rumble Motor that operates indefinitely from a single button press is not desirable from both a safety and a product lifespan point of view.

7.1.11 Stopping Rumble Motor when Controller Not in Use [Required]

The Rumble Motor must be stopped during a pause or after a reset. Also, do not start the Rumble Motor after turning on or resetting the console until there is some form of controller input. In particular, make sure not to use the Rumble Feature with the demo screen.

A Wii controller that is vibrating because the Rumble Motor is running while not in the user's hand may fall from a table or a desk.

7.1.12 Rumble Motor Run/Stop Switch Frequency [Recommended]

Do not switch between running and stopping the Rumble Motor more frequently than 100 times a second. Due to the wireless specifications, controlling the Rumble Motor at higher frequencies is difficult.

7.1.13 Hot Swapping Extension Controllers [Recommended]

Please do not have the user insert/remove extension controllers unless it is necessary. This is to reduce connector wear.

7.1.14 Frequency Band Usage when Using a Wireless LAN [Information]

The Wii Remote uses the same 2.4 GHz frequency band as some wireless LANs. The WPAD library has therefore been equipped with functionality for automatically avoiding interference.

7.1.15 Do Not Use Origin Reset Command for Extension Controller Analog Input as Game Input [Required]

Do not use commands in applications that include use of the – Button, + Button, A Button, and B Button at the same time, since this command is used to reset the origin for analog input to the extension controller. The origin can be reset by holding these buttons down for three seconds.

7.1.16 Processing Hot Swapping of Unsupported Extension Controllers [Required]

Make sure that the application operates normally when an unsupported extension controller is connected or disconnected. Focus especially on software operation when the extension controller is connected.

While an unsupported extension controller is connected, the application should display a warning message and stop the game progress or operate as if no controller is connected until the extension controller is disconnected.

In order to verify behavior when an unsupported extension controller is hot swapped, the `WPADAttachDummyExtension` and `WPADDetachDummyExtension` are provided for debugging purposes. For usage information, see the WPAD library function reference in Revolution SDK.

7.1.17 Handling HOME [Required]

Do not use HOME for any purpose other than displaying the HOME Menu.

For details about the HOME Menu, see Chapter 11 HOME Menu.

7.2 Wii Remote

7.2.1 Using the Relative Distance Between the Wii Remote and the Sensor Bar [Recommended]

Be sure to use a relative distance for the distance to the Sensor Bar. For example, do not use an absolute distance of the type where the remote has moved some number of centimeters closer to or farther away from the Sensor Bar.

7.2.2 ~~Handling HOME [Required]~~ [Deleted]

(Moved to section 7.1 All Controllers.)

7.2.3 Disabling Pointer for Applications Without Pointer Function [Required]

To save battery power in the Wii Remote, turn off power to the Pointer in applications other than the HOME menu that do not use the Pointer functionality.

For details about the HOME Menu, see Chapter 11 HOME Menu.

7.2.4 Disabling Pointer for Applications with Pointer Function [Recommended]

Even if the application does use the Pointer function, turn off power to the Pointer in scenes that do not use the Pointer. This is to preserve battery power in the Wii Remote.

It is not necessary to disable and enable the Pointer frequently. The Pointer can be disabled and enabled using the `WPADControlDpd` function in the WPAD library. When using the KPAD library, it can be disabled using the `KPADDisableDPD` function and enabled using the `KPADEnableDPD` function.

Note: The two latter functions internally call the `WPADSetDataFormat` function, so if the Pointer is turned on and off, the data format will be changed. As a result, the burn-in reduction feature will be rendered inoperative.

To read about the burn-in reduction feature, see 6.21 Applying Screen Burn-In Reduction Setting [Required] in the *Wii Video Interface Library (VI)* manual and the *Video Interface Library* section of the *Revolution Function Reference Manual.*.

7.2.5 Prohibition of Simultaneous Use of Wii Remote Memory and Wii Remote Speaker [Required]

The Wii Remote speaker cannot be used while accessing Wii Remote memory. Stop using one before using the other.

The reason for this is that the Wii Remote speaker cannot output correctly if it is used at the same time as the Wii Remote memory.

7.2.6 Restriction on Contents of Data Written to Wii Remote Memory [Required]

Store data in Wii Remote memory in such a way that it will not interfere with game progress if data is lost.

This requirement is due to the fact that there is a possibility that data in Wii Remote memory may be overwritten when another game is played.

Only use Wii Remote memory to store data that can be recovered from data in Wii console NAND memory when data is lost, or to store temporary data that will not interfere with game progress even if lost.

7.2.7 Wii Remote Data Error Checking [Required]

When using data from Wii Remote memory, always perform error checking using a method such as checksum. Also, if an error is detected through error checking, make sure that this does not cause any lock-ups.

Example of Error Checking:

3884 bytes data + 4 bytes checksum

7.2.8 Message Display when Writing to Wii Remote Memory [Required]

When writing data to Wii Remote memory, display a message similar to CONT_03 in the *Wii Message List* to prevent the user from inserting or removing an extension controller while writing to Wii Remote

memory.

This is due to the fact that some data being written to Wii Remote memory may be corrupted if the extension controller is removed or inserted during a write operation.

7.2.9 Message Display when Loading Wii Remote Memory [Recommended]

When reading data from the Wii Remote memory, display a message similar to CONT_03 in the *Wii Message List* to prevent the user from inserting or removing the extension controller while reading Wii Remote memory.

Data read from Wii Remote memory may be corrupted if the extension controller is inserted or removed during a read operation.

7.2.10 Error Handling when Writing to Wii Remote Memory [Required]

Be sure to perform verification after writing data to Wii Remote memory. If an error occurs during verification, attempt to rewrite as many as three times.

If rewriting to the Wii Remote memory fails three times, then the Wii remote control is physically damaged. In this case, display the CONT_04 error message exactly as written in the *Wii Message List*.

7.2.11 Error Handling when Reading from Wii Remote Memory [Required]

If an error occurs while reading data from Wii Remote memory, either attempt to reread up to three times or use error correction.

If rereading from the Wii Remote memory fails three times, then the Wii remote control is physically damaged. In this case, display the CONT_04 error message exactly as written in the *Wii Message List*.

7.2.12 Data Compatibility for the Same Application Using Wii Remote Memory [Required]

Maintain compatibility of the save data when releasing post-release updates to game programs.

7.2.13 Prohibition of Operations Dependent on Wii Remote Memory Access Speed [Required]

The Wii Remote memory's access speed depends on various external factors, such as signal conditions. Make sure that there are no memory access speed-dependent operations that may result in an application malfunction.

7.2.14 Handling Data Corruption in Wii Remote Memory [Recommended]

When an error is detected using a checksum of the data in Wii Remote memory or by other such means, or if the WPADReadGameData function returns WPAD_ERR_BROKEN, display a message similar to CONT_05 in the *Wii Message List*.

After displaying the message, it is okay to overwrite the data in Wii Remote Memory.

7.2.15 Displaying a Message when Overwriting Wii Remote Memory [Required]

When overwriting data from another game in the Wii Remote memory, display a message similar to CONT_06 or CONT_07 in the *Wii Message List*.

7.2.16 Continuous Play on the Wii Remote Speaker [Recommended]

If data is played from the Wii Remote speaker for more than eight continuous minutes, the sound may crackle or skip. To prevent sound from crackling or skipping, we recommend implementing one of the following two measures within eight minutes after sound starts playing.

- Send WPAD_SPEAKER_ON with the WPADControlSpeaker function, and then restart the speaker.

- Do not send sound to the Wii Remote for one second or more.

Crackling and skipping is caused because the speed of sound processing on the Wii console and Wii Remote do not match exactly for the individual hardware. When the Wii console speed is faster, crackling occurs; when the Wii Remote speed is faster, skipping occurs.

7.2.17 Processing when No Output Occurs on the Wii Remote Speaker [Recommended]

For scenes where sound is not output from the Wii Remote speaker for a period of time, we recommend stopping the speaker by sending `WPAD_SPEAKER_OFF` with the `WPADControlSpeaker` function.

However, if the speaker is frequently started and stopped for scenes with sound frequently output, Wii Remote operations may be affected. It is acceptable to choose not to frequently start and stop the speaker in these situations. We recommend that in place of starting and stopping you leave the speaker on and turn `MUTE` ON/OFF by sending `WPAD_SPEAKER_MUTE_OFF` and `WPAD_SPEAKER_MUTE` with the `WPADControlSpeaker` function.

7.2.18 Speaker Sound Quality Under Poor Signal Conditions [Recommended]

When playing sound from the Wii Remote speaker, we recommend checking the signal conditions using the `WPADCanSendStreamData` function. This is to prevent sound from crackling due to specifications when signal conditions are poor.

Applications cannot detect crackling sounds. The `WPADCanSendStreamData` function returns whether crackling is occurring at the time when sound is sent to the Wii Remote. When `TRUE` is returned, sound can be sent without crackling. However, if data is not sent when `FALSE` is returned, note that that portion of sound is omitted (skipped). In addition, if sound is not sent from the result of the return value of the `WPADCanSendStreamData` function, there is no need to restart the speaker in regards to the skipping.

For specific use of this function, see the WPAD sample demo in the Revolution SDK.

7.2.19 Displaying a Message Notifying the User that Battery Charge is Low [Recommended]

When informing the user that Wii Remote battery power is low, display either an icon or a message similar to CONT_09 in the *Wii Message List*.

While it is true that the user can confirm the battery charge for the Wii Remote at any time using the HOME Menu, the application might encounter problems if the battery for the Wii Remote were to suddenly become depleted.

7.2.20 Displaying a Message Notifying Users that Wii Remote Transmissions Have Terminated [Recommended]

When Wii Remote communications have been interrupted, display a message similar to CONT_10 in the *Wii Message List*.

This is to prevent a situation where, for example, during an action-intensive game, communication from the Wii Remote is suddenly terminated, and the game progresses without the user understanding the reason for the sudden loss of control.

7.2.21 Application of Pause Function at Termination of Wii Remote Communication [Information]

It is preferable to pause the application when communication from Wii Remote is terminated.

7.2.22 Specification of Simple Pairing Procedure [Required]

Applications supporting simple pairing must perform this synchronization when the 1 Button and the 2 Button are pressed simultaneously.

The same buttons are used as the simple pairing in the HOME Menu to avoid user confusion.

7.2.23 Message Displayed During Simple Pairing [Required]

Applications supporting simple pairing must display a message similar to CONT_11 in the *Wii Message List* when performing simple pairing.

7.2.24 ~~Display Format for Simple Pairing [Required]~~ [Deleted]

(This was deleted so that the term "One Time Mode Synchronization" is only mentioned in the *Wii Terminology*, like other terms.)

7.2.25 Use of Function WPADSetSpeakerVolume Prohibited [Required]

Adjust the Wii Remote speaker volume from the application by converting waveform data sent to the Wii Remote instead of using the `WPADSetSpeakerVolume` function.

7.2.26 Settings for Automatic Disconnection of Wii Remote Communications [Recommended]

The time limit for automatically disconnecting communication with the Wii Remote should not be changed from the standard five minutes unless absolutely necessary. If you do change it, do not set it to an unnecessarily long time.

This is to keep battery consumption of the Wii Remote to a minimum.

For example, in cases where a movie lasting five minutes occurs within a game and there will not be any user operations for a long time, there is no problem with extending the time limit in light of the length of the movie. However, we recommend restoring the time limit to five minutes after returning to the game.

The time limit for automatic disconnection can be set with `WPADSetAutoSleepTime`.

7.2.27 Handling when a Wii Remote is Connected to a Player Number Not Used by the Application [Recommended]

If Wii Remotes are connected to player numbers not being used, we recommend they be disconnected immediately in order to keep the battery consumption of these Wii Remotes to a minimum.

For example, assume that two Wii Remotes can be used by a particular application. If Wii Remotes with player numbers 3P or 4P are connected while this application is running, you should disconnect them immediately.

When returning to the application from the HOME Menu, a Wii Remote may have been connected in the HOME Menu, so verify the Wii Remote connection count restrictions.

7.3 Nintendo GameCube™ Controller

Applications that support the Nintendo GameCube Controller must support the items in this section.

7.3.1 ~~Application Support for the Nintendo GameCube Controller [Information]~~ [Deleted]

(The content of this item was moved to the section 7.3 introduction because it is a description that relates to the entire section.)

7.3.2 Use of Origin Reset Command as Game Input Prohibited [Required]

To avoid resetting each stick and trigger when performing actions necessary in the application, do not create a command involving simultaneous pressing of the X Button, Y Button, and START/PAUSE in the application.

The Nintendo GameCube Controller is designed to reset the origin for each stick and trigger by holding down the X Button, Y Button, and START/PAUSE simultaneously for three seconds.

7.3.3 Support for Origin Reset Command [Required]

To avoid user confusion, make sure the application will not react in an unexpected manner when the Control Stick and the R Button, L Button, and C Stick are reset.

7.3.4 Hot Swapping Support for Nintendo GameCube Controller Socket Devices [Required]

Support hot swapping of the Nintendo GameCube Controller Socket being used.

The user may insert or remove the Nintendo GameCube Controller in the middle of running the application. Execution of the Nintendo GameCube Controller origin reset command and Nintendo GameCube WaveBird™ Wireless Controller channel switching will also result in a state similar to that of Controller hot swapping.

Call the `PADReset` function for sockets that return the error value `PAD_ERR_NO_CONTROLLER`.

7.3.5 Frequency of Checks for Nintendo GameCube Controller Hot Swapping [Recommended]

When a Nintendo GameCube Controller is inserted, make sure that there is no delay before controls are enabled in the application.

Otherwise, this would temporarily impede players' game progress.

7.3.6 Nintendo GameCube WaveBird Wireless Controller Support [Information]

The WaveBird can be treated the same as a standard Nintendo GameCube Controller.

Note that the WaveBird communicates over the 2.4GHz band, which may affect the following:

- Wi-Fi communication by the Wii console
    - o Communication when WiiConnect24 is turned on
    - o Communication by a Nintendo Wi-Fi Connection-compatible application

- Wii Remote communication

7.3.7 Stopping Access to the Optical Disc Drive in Scenes Where the User is Urged to Insert or Remove the Nintendo GameCube Controller [Required]

Stop access to the optical disc drive in scenes where the user is urged to insert or remove the Nintendo GameCube Controller. This is because, for example, if the user knocks over the Wii console during removal or insertion and jolts it, the disc or optical disc drive may be damaged.

7.4 Wii Zapper

Applications that support the Wii Zapper must support the following items.

(For applications that support the Wii Zapper and the Wii Wheel, a soon-to-be released document tentatively named the Wii Controller Hardware Overview will describe precautions for game design.)

7.4.1 Displaying the How to Hold the Wii Zapper Screen [Required]

Display the How to Hold the Wii Zapper screen immediately after the Strap Usage screen.

For details, see the How To Hold Wii Zapper Screen Package.

8 Wii Console NAND Memory

8.1 Restrictions on Wii Console NAND Memory [Required]

The limits shown in Table 8-1 are placed on the size of the home directory and the `/tmp` region that can be allocated by each application. Do not exceed these upper limits.

Table 8-1   Usage Limitations for Wii Console NAND Memory

Directory	Max Size	Number of Files/Directories
Home directory	16 MB	32
/tmp	40 MB	64

These restrictions are necessary because the Wii console NAND memory is shared with other applications.

Save banner files are also subject to size and file number restrictions.

8.2 Limits on Frequent Rewrites [Required]

To reduce wear on the Wii console NAND memory, avoid unnecessary calls to functions that update the FAT (functions like `NANDCreate`, `NANDDelete`, and `NANDClose`). For example, avoid excessive writing to Wii console NAND memory, like saving every second or every time the character moves.

If you are using an auto-save feature, set the frequency to once a minute at the most. If there are minigames that take less than a minute and are played repeatedly, this rule cannot be followed. In such cases, it is all right to save with a frequency of more than once a minute sometimes, just as long as the average frequency for your overall application is once a minute at the most.

If you are using the `NANDSimpleSafe` series API for auto-saving, set the frequency even lower at once every 4 minutes at the most.

A list of functions that update the FAT will soon be released.

8.3 Error Handling [Required]

Always include an error handling routine (including error display) for the Wii console NAND memory.

For details on error handling, see "NAND Processing Sequence" in the *Revolution Function Reference* manual.

For error types and response policy, see "Handling Result Codes" on the "Introduction" page of the NAND API description in the *Revolution Function Reference* manual.

8.4 Write Operation Display [Required]

When writing save data larger than 1 MB to the Wii console NAND memory, display a message or an icon indicating the data write operation is underway. This is to prevent the user from unintentionally destroying data important to the user by turning off power or resetting the Wii console while writing data to the Wii console NAND memory.

For example, display a message similar to NAND_09 in the *Wii Message List* during the write operation itself.

It is okay to display the message for a longer time than it actually takes to perform write operations to make the message easier for the user to comprehend.

If the file is a temporary file or a saved file that would not create a grave disadvantage for the user even if the data contents were lost, then this display is not required.

8.5 Data Compatibility Within the Same Program [Required]

Maintain save data compatibility even if there is a change made to the program after mass production.

8.6 Applications Dependent on Wii Console NAND Memory Access Speed Prohibited [Required]

Make sure there is no program malfunction or crash because of a dependency on access time to the Wii console NAND memory.

This is required because there is performance variance between individual devices. Performance will also decrease as the device ages.

8.7 Handling Reset While Writing [Required]

If a reset is performed in the middle of a write operation, writing will be interrupted, causing a potential loss of data. Always verify the end of the write operation before performing a reset.

8.8 Data Unit for Users [Required]

When notifying users of information such as available memory or the size of files, use units of 128 KB and use the term "block(s)."

When notifying the user of the required number of block(s), be sure to total the size of files required by the game in file system block units, and count the equivalent number of bytes in 128 KB blocks and display. Be sure to round up any fractional part that is less than 128 KB.

For example, assume that a given game uses the following files.

- Banners/icons: 32 KB
- File 1:   128 KB
- File 2:   16 KB

In this case, the total size is 176 KB. In blocks, this is one block with a remainder of 48 KB. In this case, round the fractional part up and display "2 blocks" for the size.

In addition, when informing the user of the available memory, please display the number of bytes of available memory in blocks of 128 KB. If the available memory is less than 128 KB, then use a rounded numerical value.

Be sure to note that there is a difference between the block unit used for users and the "file system block" (equal to 16 KB) used by the Wii console NAND memory file system.

8.9 Checking the Number of Free inodes and Amount of Free Memory Using NANDCheck[Async] [Required]

Before creating a new file or directory in the home directory, always check to see if it is possible to create the file or directory by calling the `NANDCheck[Async]` function. The `NANDCheck[Async]` function determines whether there are enough free inodes and free memory in the file system.

8.10 Timing for Checking the Number of Free inodes and the Amount of Free Memory [Required]

Be careful when you call the `NANDCheck[Async]` function to check the amount of available memory and the number of available inodes. The following type of problem may arise if you check only just before saving data without paying particular attention to the exact timing of the check.

Assume that a user is playing the time attack mode of a certain racing game. After the player achieves the best time, he or she attempts to save that best time and ghost data but gets the following message:

"There is no more available space in Wii console NAND memory."

In this scenario there will be no trace of the best time and the ghost data.

Take care that this type of problem does not arise by checking resource availability using the `NANDCheck[Async]` function and creating needed files ahead of time.

8.11 Notification when the Number of inodes or the Amount of Free Memory is Insufficient [Required]

If the `NANDCheck[Async]` function finds that there are not enough available inodes or available memory, be sure to notify the user by displaying an appropriate message. For example, display a message similar to NAND_02 or NAND_03 in the *Wii Message List*.

There are situations where there is no need to inform the user why the save could not be done. If there are files that have no effect on gameplay, even if they are not created or saved, and the application will not notify the user that such files have been created, there is no need to display a message.

The `OSReturnToDataManager` function has been prepared for times when users want to reorganize the contents in Wii console NAND memory. This function exits the executing application and moves to the Wii Menu's data management screen.

If there is insufficient free space or not enough inodes, you can ask the user and move to the Data Management screen in the Wii Menu in your specification.

If you do not plan on incorporating a move to the Wii Menu's Data Management screen, be sure to display a message like NAND_10 from the Wii Message List.

When there is insufficient memory available and an insufficient number of available inodes at the same time, give higher priority to the message about there being insufficient memory available.

For details about `NAND_RESULT_MAXBLOCKS` and `NAND_RESULT_MAXFILES`, see section 8.26 Support for NAND_RESULT_MAXBLOCKS and NAND_RESULT_MAXFILES [Recommended].

8.12 Do Not Change Created File Size [Required]

In general, do not change the size of files once created. This is to prevent insufficient memory or applications becoming unable to proceed due to increase in file size while an application is running. Be sure to create files that are large enough to begin with.

Contact support@noa.com in the case of applications that are impractical without variable file size.

8.13 Handling Corrupted Files [Required]

If a NAND function returns the result code `NAND_RESULT_ECC_CRIT` or `NAND_RESULT_AUTHENTICATION`, or if the size of the save file differs from the size expected by the application program, display a message similar to NAND_07 in the *Wii Message List* indicating that data has been corrupted and that the file cannot be used. Allow the user to delete the old files and to create new ones.

8.14 Handling Corrupted Wii Console NAND Memory [Required]

Be sure to display the NAND_08 message exactly as written in the *Wii Message List* if
NAND_RESULT_CORRUPT is returned as the result code for a NAND function.

Also, disable the controller input, RESET, and Power Button while the error message is displayed.

Since NAND_RESULT_CORRUPT indicates a state where even a successful reset is not guaranteed,
place higher priority on the error message display instead of performing a reset or shutdown.

8.15 Saving Data During the Shutdown Process [Recommended]

If pressing of the Power Button has been detected, applications must quickly begin shutdown
processing. If the application needs to save some data (such as the overall play time of the
application) in Wii console NAND memory during shutdown, make sure that saving can be completed
within one second.

For details, see section 10.2 Wii Power Button [Required].

8.16 Prohibition of Creating Files or Directories Without Owner Read Permission [Required]

Files and directories need to be configured with at least read permission for the owner
(NAND_PERM_OWNER_READ).

With Revolution SDK version 2.3 patch 5 and later, a function call to create or change the permission
of a file or directory will fail if there is no owner read permission.

8.17 Prohibition of Frequent Read Access [Required]

Due to the characteristics of Wii console NAND memory, a Read operation repeated over several
hundred thousand times without any Erase or Write operations for a given region may corrupt the data
in the block that includes that region. To avoid this problem, first load a set amount of data to
MEM1/MEM2, and then access this loaded data when data stored in Wii console NAND memory needs
to be read repeatedly.

Specifically, accessing a small region of a few kilobytes at a frequency of approximately every frame
or every second is prohibited. For example, pay close attention to programs that perform the following
types of operations.

- Read access for every character in a font displayed

- A periodic Read access every frame or every second

- Read access every time a character walks

8.18 Save Banner File Creation

Implement the following items so that the appropriate save banner can be displayed for the created save data.

8.18.1 Timing of Creation [Required]

Save banner files should be created at the very end of the process of creating the minimum required files, from a state in which there are no files in the home directory (for example, at the first application startup.)

The Save Data Management screen of the Wii console deletes any save data without save banner files in the save data directory. This is to prevent file inconsistency from invalid files that may be accidentally created without a save banner file (from a sudden power outage, for example). If the necessary files were created after the creation of a save banner file, the save data screen would display the save data as valid, but the data would be invalid when the application was restarted, potentially confusing the user.

Also, if only a save banner file were found in the save data directory, it could not be copied to the SD Card in the Wii console Save Data Management screen. To avoid these problems, the save sequence needs to follow the steps below.

1. Begin in a state in which there are no files in the home directory (for example, at the first application startup)
2. Create save data/directory
3. Create save banner file
4. Start application

8.18.2 Check the Save Banner File [Required]

Every time the application starts, check to confirm that the save banner file was created as intended by the application (for example, that it has not been corrupted). If the save banner file has not been created properly, the application must create the save banner file again.

If the save banner file was not created as intended (due to a power failure or some other reason), the Wii console's Data Management screen will not display save data correctly, and the user will not know which save data belongs to which application.

8.19 Precautions for Creating Save Data [Information]

See the *NAND Library Function Reference* manual for the methods and precautions for avoiding data loss and corruption at creation of save data.

8.20 Save Data Comments [Required]

Always place the game title in the first line of save data comments. A comment of your choice may be placed in the second line.

If the comment does not fit in the second line, the first line may start with the game title and be in the format "game title + any text."

The comments must be set in the Unicode format. A null character (¥0) is treated as a termination character, but to prevent extra characters from remaining after the comments, always be sure to fill the remaining area with null characters.

A WBF (Wii Bitmap Font) Gothic font (WBF1) is used to display the comments. The usable characters in WBF1 are as follows (all characters other than extended characters):

- Alphanumeric characters (roman characters)
- Greek alphabet characters
- JIS half-width characters
- JIS non-kanji characters
- JIS level 1 kanji
- JIS level 2 kanji
- Roman numerals

In the structure, 32 characters worth of area is allocated per line, but the number guaranteed for display is 20 characters. Always insert null for characters 21 and after.

8.21 Save Data Location Specification [Required]

A save banner file should always be created in the save data directory prepared by the system (/title/<title-id(Low)>/data) with the filename banner.bin.

An application creates only a single save banner file. Make sure that this is created in the save data directory with the specified file name.

8.22 Prohibition of Using Reserved Names [Required]

The names noerase and notransfer are reserved by the system. Creating a file or a directory in the save data directory with these names is prohibited.

8.23 Precautions Regarding the nocopy Directory [Required]

Do not create any data whose loss may critically affect the game progress in the nocopy directory.

When such data is placed in the nocopy directory, the original data will become unrecoverable if the

user erases the data in the Wii console NAND memory assuming that the backup save data is copied to the SD Card.

8.24 Save Data Tampering Protection by the Wii Console [Information]

When save data is being copied to the SD Card in the Wii console Data Management screen, a data tampering protection will be applied. In general, there is no need to implement any tampering protection for save data saved to the Wii console NAND memory from the application side. However, security is not guaranteed for the future, so data tampering processing can also be implemented by the application in order to ensure better security.

8.25 Prohibition of Usage Restrictions on Save Data Due to Wii Console-Specific Information [Required]

Creating save data that uses information unique to each Wii console (for example, MAC addresses) and that can only be used by the Wii console that has that unique information is prohibited.

This is to prevent save data from becoming unusable if a module or the Wii console is replaced during repair.

8.26 Support for NAND_RESULT_MAXBLOCKS and NAND_RESULT_MAXFILES [Recommended]

When either NAND_RESULT_MAXBLOCKS or NAND_RESULT_MAXFILES is returned, the operation of the system is not guaranteed. We recommend that you quickly display either the NAND_05 or NAND_06 message depending on the return value, and do not return to the application program's original processing routine. When used, these messages must be displayed exactly as written in the *Wii Message List*.

8.27 ~~[N. American Version Only] Considerations for Save Banners that Do Not Have Alpha Values Applied~~ ~~[Required]~~[Deleted]

(Deleted because changes to the Wii Menu make this unnecessary.)

9 Nintendo GameCube Memory Card

Do not use the Nintendo GameCube Memory Card.

The CARD library will be deleted from the soon-to-be-released Revolution SDK.

9.1 ~~Nintendo GameCube Memory Card Compatible Application [Information]~~[Deleted]

9.2 ~~Use of Write Functions Prohibited [Required]~~[Deleted]

9.3 ~~Programs Must Not Depend on the Access Speed of Nintendo GameCube Memory Cards [Required]~~[Deleted]

9.4 ~~Support for High-Capacity Nintendo GameCube Memory Card [Required]~~[Deleted]

9.5 ~~Errors Displayed when Memory Cards Cannot be Used [Required]~~[Deleted]

9.6 ~~Compatible Slots [Recommended]~~[Deleted]

9.7 ~~Memory Card Menu in Application [Required]~~[Deleted]

9.8 ~~Limit on Accessible Memory Card Files [Required]~~[Deleted]

10 Reset and Shutdown

10.1 Wii Console RESET [Required]

If RESET is pressed while an application is running, immediately perform the pre-processing for reset, and perform the reset operation once this pre-processing is complete. Also, do not perform any operation other than a reset when RESET is used.

However, see section 8.7 Handling Reset While Writing [Required] regarding writing to the Wii console save memory.

For details, see the *Wii Console Reset and Shutdown Programming Manual* and the *Revolution Optical Disc Drive Library (DVD)* manual.

10.2 Wii Power Button [Required]

If the Power Button on the Wii console or Wii Remote is pressed while an application is running, immediately perform pre-processing for shutdown, and perform the shutdown operation once pre-processing is complete. Also, do not perform any operation other than a shutdown when the Power Button is pressed.

Also, holding the Power Button of the Wii console for approximately four seconds will cause the Wii console to turn off by means of a hardware shutdown.

For details, see the *Wii Console Reset and Shutdown Programming Manual* and the *Revolution Optical Disc Drive Library (DVD)* manual.

10.3 Resetting the Nintendo GameCube Controller Origin During Reset Operations [Required]

Always perform an origin reset for the Nintendo GameCube Controller when a reset is performed in an application that supports the Nintendo GameCube Controller.

The user may press RESET for a correction when the origin goes out of alignment.

Here, the reset operation refers to the reset operation performed with either RESET on the Wii console or the Reset button on the HOME Menu.

When using the function `OSRestart` for reset, use the function `PADRecalibrate` to reset the origin. If `OSReturnToMenu` is being used to return to the Wii Menu, there is no need to perform an origin reset because the Nintendo GameCube Controller cannot be used to operate the Wii Menu.

The origin for the Nunchuk, Classic Controller, or WaveBird will not be reset when the origin reset is performed from the Wii console.

10.4 Use of the OSRebootSystem Function [Recommended]

Avoid the use of `OSRebootSystem` as much as possible.

It takes more time for an application to restart with the `OSRebootSystem` function because the boot screen is displayed.

10.5 ~~Required Preparations Before Using the Reset/Shutdown Functions~~ ~~[Required]~~[Deleted]

(The content of this section was deleted because it was redundant with sections 10.1 Wii Console RESET [Required] and 10.2 Wii Power Button [Required].)

10.6 Restrictions on Operations for Resets [Required]

Be sure to perform a reset only when RESET on the Wii console is pressed or when the reset option is selected using the HOME Menu.

11 HOME Menu

11.1 Using the HOME Menu Library [Required]

When adding a Home Menu to the application, use the HOME Menu library (HBM Library).

Changing the HOME Menu library and the included data (screen layout, images, sound, etc.) is prohibited.

11.2 HOME Menu Display Triggers [Required]

Only display the HOME Menu immediately after HOME is pressed. Do not display it under any other circumstances. Do not display a confirmation message such as "Do you want to go to the HOME Menu?"

11.3 Display Language for the HOME Menu [Required]

The language selected in the system settings should be used as the HOME Menu language, regardless of the language used in the application.

For example, if the application is for the North American market and uses only English, but French is selected in the system settings, display the HOME Menu in French.

A list of language options for each market is shown in Table 11-1   Selectable Languages for Each Market.

Table 11-1   Selectable Languages for Each Market

Market	Language Selection
Japan	Japanese
North America	English, French, Spanish
Europe/Australia	English, French, Spanish, German, Italian, Dutch

11.4 Screen Ratio of the HOME Menu [Required]

If the application supports 16:9 screen ratio, the screen ratio of the HOME Menu should also follow the setting of the Wii system.

For details, see section 6.16 Settings and Display for Applications that Support 16:9 Screen Ratio Display [Required].

11.5 ~~Using the HOME Menu from All Paired Wii Remotes~~ ~~[Recommended]~~

(Deleted in accordance with guideline revisions.)

11.6 Support for HOME Menu by Classic Controllers [Recommended]

Applications supporting Classic Controllers should support the following:

- Display the HOME Menu when HOME on either the Classic Controller or the Wii Remote is pressed.

- Allow control in the HOME Menu from both the Classic Controller and the Wii Remote.

11.7 Application Processing During HOME Menu Display [Recommended]

Since the user cannot perform any application-related operations while the HOME Menu is displayed, be sure to consider the issue of timing when displaying the HOME Menu so that this does not inconvenience the user.

For example, in the case of an action game, make sure that the game does not continue while the HOME Menu is displayed, resulting in damage from an opponent or loss of remaining time.

11.8 Restrictions on Rumble Feature

11.8.1 Restrictions Due to the Rumble Setting of the HOME Menu [Required]

If the application has its own controller Rumble Feature option setting, and the Rumble setting in the Wii Remote Settings screen of the HOME Menu is set to Off, the controller's Rumble Feature should not be activated regardless of the application's own option setting.

The Rumble Feature for the controller should only be used if the Rumble setting in the Wii Remote Settings screen of the HOME Menu is set to On, for example; if the return value of the `WPADIsMotorEnabled` function is TRUE.

11.8.2 Prohibition of Changing the Rumble Setting of the HOME Menu [Required]

Changing the Rumble setting of the Wii Remote Settings screen in the HOME Menu from the application is prohibited.

If the Rumble setting of the Wii Remote Settings screen in the HOME Menu is set to Off, for example, if the return value of the `WPADIsMotorEnabled` function is FALSE, running the `WPADMotorEnable(TRUE)` function and then calling `WPADSaveConfig` will change the Rumble setting of the Wii Remote Settings screen in the HOME Menu to On. Do not use the `WPADEnableMotor` and `WPADSaveConfig` functions in Master ROM versions. They may only be used for debug.

11.9 Speaker Initialization After Leaving HOME Menu [Required]

After returning from the HOME Menu to the application, the state of the Wii Remote speaker is undefined. Always be sure to initialize the speakers after returning to the application.

11.10 Prohibition of Wii Remote Speaker Use During HOME Menu Startup [Required]

While the HOME Menu is starting up, the application must not operate the Wii Remote speaker.

11.11 Display of HOME Menu Disabled Icon [Recommended]

If the HOME Menu cannot be displayed, display the specified HOME Menu Disabled Icon when possible when the Home Button is pressed.

There is no need to display the HOME Menu Disabled Icon when the Strap usage screen is being displayed, a fatal error has occurred, or when a user would not feel anything is wrong even if the Wii Remote input has no effect.

11.12 Processing when Quitting the HOME Menu [Required]

When returning from the HOME Menu to the application, get the button pressed when quitting the HOME Menu using `HBMGetSelectBtnNum`. When the button is either Wii Menu or Reset, perform the corresponding process immediately.

After returning to the application upon selection of the Wii Menu button or Reset button, do not display a verification message (such as "Would you like to Quit?"). Also, do not restore the volume of the application (leave it on mute).

11.13 Sound Volume when Entering and Exiting the HOME Menu [Required]

When entering the HOME Menu, set the sound volume to 0 on the application side. When exiting from the HOME Menu, initialize sound effects on the application side when the HOME Menu ends, and restore the sound volume.

11.14 Displaying Error Messages for Errors that Occurred While in the HOME Menu [Information]

If the generation of an error is detected while displaying the HOME Menu, it is acceptable to display an error message immediately after returning to the application from the HOME Menu.

For example, if the HOME Menu is displayed and the disc is ejected while data is being read from the disc, you can display a *No disc* error message as soon as the process returns to the application.

However, if an error occurs that might make even rebooting the Wii console impossible (such as a *Fatal error* regarding the disc drive or NAND_RESULT_CORRUPT regarding Wii console NAND Memory), display the error message in the HOME Menu.

12 Icons and Banners

12.1 Icon Animation [Required]

Always use an animation for the Wii Menu channel list icon so that it will not be a stationary image.

The reason for this is to make the Wii Menu resemble a television channel interface.

12.2 Use of Text Boxes for Icon Layout [Required]

If using text boxes for the icon layout, be sure to use the bitmap font included in the Icon and Banner Creation Tool. The use of any other fonts is prohibited.

12.3 Icon and Banner Data Size Restriction [Required]

The data size for icons and banners must be 512 KB or less after being compressed with the archiving tools. Icon resources, banner resources, and banner sound data will be compressed and stored into banner data.

12.4 Title Display in the Banner [Required]

Always display the software title name in the banner.

The title name can be displayed as a texture or through a text box, and the display size may be set freely. The title name does not need to be displayed at all times, and may become hidden as a part of the presentation.

We recommend using the language set in the Wii system language setting for the title name display.

For details on languages used in banners, see section 3.5 Switching Languages in the *Icon and Banner Specification.*

12.5 Banner Animation [Information]

Animation is not required for the banner displayed when a Wii Menu channel is selected. Still images are acceptable.

12.6 Use of Text Box for Banner Layout [Required]

If using text boxes for the banner layout, make sure to use the bitmap font included in the Icon and Banner Creation Tool. The use of any other fonts is prohibited.

12.7 Display of Trademarked or Copyrighted Names in the Banner [Information]

The Channel banner displayed in the Channel Preview Screen of the Wii Disc Channel may include trademarked or copyrighted names such as company logos.

This can shorten the user wait time by removing the necessity of trademarked displays in the application startup sequence.

As a reference, titles developed by Nintendo will display their copyrights in the application title screen, but trademarks will generally be displayed in the banner to shorten the transition time to the application title screen.

A minimum of one second is guaranteed from the completion of icon selection screen zoom effect (or transition effect from the neighboring banner) to the start of the application title screen transition fadeout. Ensure that if the copyright/trademark display is presented here, the information is not faded in and then out, because it could be skipped if the parental controls are activated.

However, the above suggestion is optional, so the trademark or copyright information may still be displayed in the application startup sequence or title screen as traditionally done. Determine the display method based on company policy or what is appropriate for the application.

Trademark or copyright information may also be displayed in both the banner and the startup sequence/title screen. In such a situation, please make sure that the display content between the two stays consistent.

12.8 ~~[N. American Version Only] Consideration for Non-Application of the Banner Alpha Value [Required]~~[Deleted]

(This description is about save banners, so the content was moved to Chapter 8 Wii Console NAND Memory**Error! Reference source not found.**.)

12.9 Banner Sound Setting [Required]

Always set a banner sound to ensure that the screen is not silent during the banner display.

The reason for this is to make the Wii Menu resemble a television channel interface.

12.10 Displaying the Title via the Banner Data Header [Required]

In the banner data header, do not specify an arbitrary string that is irrelevant to the title of the application.

In the first row of the banner data header, enter a title that basically suggests the application name. It does not need to be exactly the same as the actual title of the application. In the second row, specify something like a subtitle.

Note that the banner data header will be referenced by "Today's Accomplishments" in the Wii Message Board.

13 Communications with Nintendo DS

13.1 General

13.1.1 Adherence to the Use of the Library [Required]

When communicating with the Nintendo DS (referred to as DS throughout the rest of this document), be sure to use the libraries provided by Nintendo.

13.1.2 ON and OFF State of MP Communications [Information]

The ON state of MP Communications refers to the state in which wireless signals can be sent or received, or the state in which wireless signals are actually being sent or received for the purpose of wireless communications with the DS. To be specific, this state amounts to the period between the execution of the `MPStartup` or `MPDLStartup` functions and the `MPCleanup` or `MPDLCleanup` functions. (MP Communications will be in the OFF state during all other intervals.)

This is different from the situation with the DS in that MP Communications can be turned ON without the user's permission and with arbitrary timing, and there is no problem with turning them ON automatically immediately after application startup.

13.1.3 Downloading Programs [Required]

Sending programs (native ARM code) to the DS and running them through means other than the MPDL and MPFS libraries is prohibited.

13.2 MP Communications with the DS

13.2.1 Message Display for Severed Connections [Required]

When a severed connection makes it difficult to continue the game, display a warning message such as "Cannot receive data" when the severed connection is detected.

For example, if the connection is severed for only one of several connected DS systems, a warning message must be displayed on the DS with the severed connection. But if there is no overall effect on the application's use by other players, no warning messages need be displayed to other DS systems or Wii consoles that are still connected.

If there is a special reason for not wanting to notify the user of a severed connection (or that MP Communications associated with the severed connection were carried out), there are special cases in which hiding messages (or falsely displaying them) is allowed.

If you have such plans, contact support@noa.com before proceeding.

13.2.2 Single-Transmission Data Size for MP Communications [Recommended]

It is recommended that the total transmission time for a single transmission of MP Communications between the parent and all children should be 5600 microseconds or less.

The shorter the transmission time, the better the communication performance will be. However, there are various disadvantages if the data size is large, including being unable to establish a wireless connection, decreased reliability of data transmission, decreased number of systems which are simultaneously transmitting in the same area, increased power consumption, decreased resistance to interference, and so forth.

In order to calculate the total transmission time, use the "Wireless Communication Time Calculation Sheet" in the *Figures and information* page in the Wireless Manager (WM) section of the *NITRO-SDK Function Reference* manual distributed with the DS development environment (NITRO-SDK).

A sample calculation is shown below:

In the case of MP Communications, once every 60 seconds for communication between one Wii console and 15 DS systems:

Wii: 128 Bytes or less

DS: 16 Bytes or less

13.2.3 Which GGID to Use [Required]

For the GGID, only use the numbers that were provided by Nintendo for each game title.

(Do not take the liberty of using numbers that have not been assigned by Nintendo.)

However, during testing or in the initial stage of game development, you may use private GGIDs. Private GGIDs are secured for testing purposes and are not assigned for each individual game title. As a result, you should realize that connections may be made from other test applications using private GGIDs, which in turn can cause bugs to occur.

Private GGIDs: 0x003FFF00 - 0x003FFFFF (256 IDs)

13.2.4 Connection with Game Software of Other Companies' Titles is Prohibited [Required]

Creating game software that can connect with game titles made by other companies is prohibited.

If you have such plans, contact support@noa.com before proceeding.

13.2.5 Connection with Different Versions of the Same Software [Required]

Make sure that communications with different versions of DS software of the same title can be carried out normally.

13.2.6 Processing when Attempting to Connect More than the Supported Number of DS Systems [Required]

Make sure that communications can function normally for the supported number of DS systems, even if more than the supported number of DS systems attempt to connect to a single Wii console.

13.2.7 Screen Burn-In Reduction Feature [Required]

For applications that use the DS as a Wii controller and do not concurrently use the Wii Remote, the screen burn-in reduction feature might be activated when the user doesn't expect it.

Consequently, reset the counter for activating screen burn-in reduction (using the `VIResetDimmingCount` function) every time there is active input from the user (button or touch panel input, valid speech input to the microphone, and so forth).

Resetting the screen burn-in activation counter under any other circumstances (that is, when there is no active input from the user to the DS system that has become an input device) is prohibited.

For details on the screen burn-in reduction feature, see section 6.21 Applying Screen Burn-In Reduction Setting [Required].

13.2.8 Dealing with Invalid or Corrupted Communications Data [Required]

Because there is a chance that data sent from DS systems will be invalid or corrupted, always code with that in mind. Refer to the specific guidelines indicated in the following subsections.

13.2.8.1 Caution About Buffer Overflows

In order to prevent the system from being taken over through the exploitation of buffer overflows, avoid using variable-length data and always execute sends and receives using fixed-length data.

For example, avoid things like acquiring and using the data size from the header portion of received data.

13.2.8.2 Validity Check for Received Data

Be absolutely sure to check that each parameter of the received data falls within the expected range.

13.3 DS Single-Card Play

13.3.1 ~~Banner Icons [Information]~~[Deleted]

(Deleted because starting with Revolution SDK Extensions 2.1, Wii games can specify their own individual icons for DS Single-Card Play.)

13.3.2 TGIDs Used by the MPDL Library [Required]

If a DS Single-Card Play child has the same TGID as the parent, the parent's information will not be updated.

For this reason, when using the MPDL library, make sure the TGID is a different value each time communications begin.

Specifically, `MP_TGID_AUTO` is entered for the TGID member of the `MPDLConfig` structure.

13.3.3 Accessing Cards During DS Single-Card Play [Required]

Allowing programs transferred to a DS via DS Single-Card Play so as to access DS Game Cards made by other companies is prohibited as a rule.

Do not access memory areas other than the backup memory, even for DS Game Cards made by your company. If you have plans that involve accessing memory areas other than the backup memory, contact support@noa.com before proceeding.

13.3.4 Characters Displayed on the DS [Required]

Restrict the strings passed to the `MPDLConfig` structure to characters that can be displayed on the DS. These strings are the publisher name, program title name, and program description displayed on the DS Single-Card Play client.

For details about the characters that can be displayed on the DS, see the character code chart in the *DS IPL Banner Guidelines* package.

14 USB Keyboard

Applications that support USB keyboards (keyboards) connected to the USB connector must support the following items.

14.1 Operation Verification on Specified Models

Be sure to verify operations on Logitech's "Classic Keyboard 200" (irrespective of the keyboard color) for the following items. Nintendo has verified operations on this model.

14.1.1 Support for Different Model Numbers According to Market and Language [Required]

The "Classic Keyboard 200" has different model numbers according to the market and language, so be careful of the application language support.

For model numbers, see the "List of Keyboards for which operations have been verified" (tentative) that is planned to be provided soon.

14.1.2 Operation of Unused Keys [Required]

Ensure no problems occur in the application due to an unused or unexpected key being pressed. Be sure to appropriately support all expected and used keys and key combinations, and ignore unused keys.

14.1.3 LED Status Control [Required]

Be sure to control the keyboard LEDs with the application. Implement the NUM LOCK and CAPS LOCK keys so the user can operate them just as on the PC keyboard. SCROLL LOCK key support is optional. The sample demo in the KBD library introduces how to control the LEDs appropriately.

Excessively changing the LEDs may affect the USB operation performance. Please limit sending LED commands to only instances when the NUM LOCK, CAPS LOCK, or SCROLL LOCK keys are pressed down.

14.2 Avoiding Game Design Forcing Extended Continuous Operation or Strenuous Activity [Information]

For keyboards, as with controllers, please avoid game designs that force extended continuous operation or strenuous activity.

For controllers, see section 7.1.2 Avoiding Game Design Forcing Extended Continuous Operation or Strenuous Activity [Information].

14.3 Prohibit Hot Swapping of the Keyboard [Required]

Do not have specifications that urge the user to hot swap while the application is running. This is to control wear of the connectors and to prevent damage to the Wii console due to hot swapping.

14.4 Processing During Startup when a Keyboard is Required [Required]

For applications that require a keyboard, if a keyboard is not connected during startup, please display USBK_01 from the *Wii Message List*, and shut down after waiting for user operation.

Appendix A Revision History of Previous Versions

Version	Revision Date	Item(s) Affected	Description
1.07	2008/01/31		◆ 2.15 [N. American Version Only] Displaying the ESRB Online Rating Notice for Applications Using Network Features [Required] Updated the heading from "[N. American Version Only] Displaying a Cautionary Notice for Applications Using Online Features [Required]". Updated contents. Updated the content also. ◆ 6.18 Support for 4:3 Aspect Ratio [Required] Mentioned that it is permissible to display static images in the margins, in a manner that resembles letterboxing, and that 6.24 Avoiding Screen Burn-In Caused by Static Display [Information] should be referenced for information on displaying static images. ◆ 6.23 Support for 16:9 Screen Ratio [Recommended] Mentioned that it is permissible to display static images in the margins, in a manner that resembles pillarboxing, and that 6.24 Avoiding Screen Burn-In Caused by Static Display [Information] should be referenced for information on displaying static images. ◆ 6.24 Avoiding Screen Burn-In Caused by Static Display [Information] Removed the pillarboxing and letterboxing examples of fixed displays that can cause screen burn-in. ◆ 8.4 Write Operation Display [Required] Updated the heading from "Write Operation Display." Removed the directive to show NAND_01 from the *Wii Message List* before starting a write operation. ◆ 14.1 Operation Verification on Specified Models Mentioned that the color of the "Classic Keyboard 200" is irrelevant.
1.06	2007/12/19	2	◆ Added the following items. 2.19 Using Firmware of Specified Versions [Required] 2.20 Time Restriction when Screen is in Static State [Required]
		3.2	◆ Changed the reference to "Wii Optical Disc Drive Guidelines".
		3.8	◆ Revised description to state that program loading is performed by the Wii Menu rather than the IPL. Updated the reference to "Wii Optical Disc Drive Guidelines."
		3.10	◆ Divided the sentence that described the reset process and shutdown process into two sentences to be more easily understood.
		4.1	◆ Clarified that when the application has separate settings, the Wii console settings are applied to these separate settings when the application is launched initially.
		6.4	◆ Deleted mention of RevolutionSDK 2.2 Patch10. Also deleted mention of applications that do not support progressive display.
		6.10	◆ Deleted mention of RevolutionSDK 2.2 Patch10. Also deleted mention of applications that do not support EU RGB 60 display.
		6.16	◆ Clarified that when the application has separate settings, the

Version	Revision Date	Item(s) Affected	Description
			Wii console settings are applied to these separate settings when the application is launched initially.
		6.21	◆ Updated.
		6.22	◆ Updated.
		6.22.1	◆ Deleted in line with changes to 6.21 and 6.22.
		7.1.5	◆ Added the case of operating an unsupported extension controller as a relevant example.
		7.1.16	◆ Changed heading from "Handling Unsupported Extension Controllers [Required]". Also mentioned that the WPADAttachDummyExtension and WPADDetachDummyExtension functions were prepared for debugging in order to verify behavior when an unsupported extension controller is connected or disconnected.
		7.2.10	◆ Clarified that the Wii Remote memory is physically damaged when re-accessing fails three times.
		7.2.11	◆ Clarified that the Wii Remote memory is physically damaged when re-accessing fails three times.
		7.2.16	◆ Restored a previously deleted item as a [Recommended] item.
		7.2.17	◆ Changed heading from "Halting Data Transfers when No Output Occurs on the Wii Remote Speaker [Recommended]." Also, separately clarified the description of processing when sound is not output for a period of time and when sound is output frequently.
		7.2.18	◆ Added a specific use example, and deleted reference to the "Wii Remote and Nunchuk Hardware Overview".
		7.2.27	◆ Updated the relevant conditions for this item. Also provided more information for the case when returning to the application from the HOME Menu.
		7.3.7	◆ This item was newly added.
		8.3	◆ The "NAND Library Error Types and Handling" table was moved to the "Introduction" page of the NAND API description in the function reference manual as "Handling Result Codes." Accordingly, the table was deleted and a reference to the new location was added. In addition, a more detailed description of the reference was made.
		8.4	◆ Clarified that the data for which damage should be prevented is important data, and added explanation regarding data that does not need to be displayed.
		8.7	◆ Changed the heading from "Reset Handling" to easily differentiate it from 10.1 Wii Console RESET [Required].
		8.8	◆ Added a caution for informing the user about required memory space.
		8.12	◆ Restructured the sentence to emphasize the concern for changing file sizes.
		8.24	◆ Added information regarding protection against tampering on the application side.

Version	Revision Date	Item(s) Affected	Description
		10.1	◆ Added that 8.7 Handling Reset While Writing [Required] should be referenced regarding handling while writing to the Wii Console save memory.
		11.5	◆ Deleted this item.
		11.13	◆ Clarified that the sound volume "setting" should be set to 0.
		13.3.2	◆ Revised so that the timing by which TGID becomes a different value changes when communications begin.
		14	◆ This section was added to support USB Keyboard.
1.05a	2007/09/27	13.3.2	Corrected heading.
1.05	2007/08/10	Revision History	Corrected errors in sections 8.13, 8.14, 8.27, 10.6, and 12.8 of the revision history for Version 1.04.
		1.2	Deleted the fact that some of the referenced documents were being updated and therefore may not have contained the information being referenced.
		1.3	Added the new section "Do Not Use Wii Development and SDK Files on Other Platforms."
		2.10	Some content that referred to "Normal Operation after the Disc Drive Motor Is Shut Off From a Period of Inactivity" was still present, so it was deleted.
		2.13	Clarified that the warning screens for which display is prohibited is limited to screens relating to Health and Safety Precautions.
		3.9	Added a note to refer to section "3.11 Message Display in Response to Fatal Errors [Required]," for details about fatal error messages.
		3.11	Added the new section "Message Display in Response to Fatal Errors [Required]."
		6.24	Added the new section "Avoiding Screen Burn-in Caused by Static Display [Information]."
		7.1.1	Added the fact that there is no need to display the message if it has been confirmed that the extension controller required by the application has already been connected.
		7.2.14	Explained the conditions in greater detail, namely that [the message should be displayed when an error is detected using a checksum of the data in the Wii Remote memory or other such means, or when the WPADReadGameData function returns WPAD_ERR_BROKEN.
		8.1	There was no caption for the table, so it was changed to "Usage Limitations for Wii System Memory."
		8.3	There was no caption for the table, so it was changed to "NAND Library Error Types and Handling."
		8.11	Added the fact that there is no need to display a message if there is no need to notify the user of the reason that the save could not be performed.
		8.14	Corrected "display a message like that of NAND_08" to "display the NAND_08 message exactly" because this section is

Version	Revision Date	Item(s) Affected	Description
			referring to NAND_08, which is marked as "Changes prohibited" in the Wii Message List.
		9.5	There was no caption for the table, so it was changed to "Errors Displayed when Memory Cards cannot be Used."
1.05	2007/08/10	10.1 10.2	These paragraphs previously directed the user to refer to the Reset and Shutdown Programming Manual for details regarding the pre-processing for resets and shutdowns, respectively. The wording has been changed so that the user should refer to this document regarding the overall content.
		11.3	There was no caption for the table, so it was changed to "Selectable Languages for Each Market."
		13	Added a new chapter, "Communications with Nintendo DS"
1.04	2007/05/31	All	Changed the organization of content, as described below. Added internal document links to the "Revision Items" column of the Japanese Version 1.04 Revision History table and the content of the precautions. Formally moved all messages to the *Wii Message List*, and added notes about referencing the *Wii Message List*.
		2.1	Changed from [Information] to [Required].
		2.3	Based on the fact that the screen displayed right after startup is the Strap usage screen, added clarification regarding the screen to be displayed. Added a reference to the Strap usage screen. Deleted the section that had referenced "Wii File Restriction" in the "Revolution Optical Disc Drive Library manual."
		2.5	Given the policy that explanations for legal rights notations will be given in each library and tool, the text stating that there were "plans to distribute a document compiling various library tool rights displays" was deleted.
		2.6	The heading was changed from "Library Version" to "Library and Tool Versions." A note about tools was added to the content.
		2.9	Added MIDI adapters as an example of libraries for which releases are prohibited.
		2.10	Updated this paragraph to ensure that fatal bugs like errant processes or freezes do not occur, even if the application is run for a long time. The heading was also changed from "Demo Screen Loop Processing" to "Guaranteeing Extended Operation of the Application."
		2.11	This paragraph merely referenced content explained in other paragraphs, so it was deleted.
		2.12	Added a list of the items in Wii console configuration. Changed from [Information] to [Required].
		2.18	The description of the size and type of font required was difficult to understand, so it was rewritten.

Version	Revision Date	Item(s) Affected	Description
1.04	2007/05/31	3.3	Changed the heading from "Infinite Loops During Multiple Data Streaming Prohibited" to "Prohibition of Non-Sequential Access Performed Continuously for Long Periods of Time." Revised the description to include that continuous, non-sequential access for long periods of time is prohibited in order to prevent shortening the life span of the drive and reduce seek noise.
		3.4	Revised the description regarding the required message to refer to DISC_06 of the *Wii Message List*.
		3.5 3.6	Shutting off the disc drive motor does not have an effect on the currently-provided development environment, so deleted these paragraphs.
		3.9 3.10	Added the following new items: • Resetting and Shutting Down in Response to Fatal Errors [Required] • Resetting and Shutting Down in Response to Disc Errors Other Than Fatal Errors [Required]
		4.1 4.2 4.3	Combined the following items into "4.1 Applying Wii System Sound Settings [Required]" • "Sound Setting [Information]" • "Application of Wii Console System Sound Settings [Required]" • "Sound Output for Applications Supporting Multiple Sound Settings [Required]"
		6.1	Added a statement about the safe frame for a 16:9 aspect ratio.
		6.2	Deleted, since its contents were redundant with the content indicated in the guidelines in the "Video" section.
		6.4 6.10	Added a statement that changing the settings is prohibited with Revolution SDK 2.2 Patch10, which contained a function for changing the items in Wii System Settings that relate to progressive display and EURGB60. Made it clear that the appropriate displays are either interlaced or progressive, and either PAL50 or EURGB60, respectively.
		6.5 6.11	Since the screen displayed right after startup is the Strap usage screen, clarified the screens to be displayed. Added a reference point for the Strap usage screen.
		6.14	Deleted MPAL.
		6.15	Added that anamorphic widescreen is recommended.
		6.16	Clarified that the appropriate display is either 4:3 or 16:9. Previously it was stated that changing items in Wii System Settings that relate to the 16:9 aspect ratio was prohibited when using Revolution SDK 2.2 Patch 10, but the function in question is not being provided, so the statement was deleted.
		6.19	Made a correction to refer to the "Video Interface Library" Programming Manual instead of the "Video Interface Library" Programming Guidelines.

Version	Revision Date	Item(s) Affected	Description
1.04	2007/05/31	6.23	Added the reason that a 16:9 screen aspect ratio is recommended to the description. Added a reference to the "Video Interface Library" Programming Manual.
		7.1.1	Deleted the example message. Added a reference to CONT_01 and CONT_02 in the *Wii Message List*.
		7.1.16	Added to the description that with Revolution SDK 2.4 Patch 6 and later, functions have been added for changing the device type of the Wii Remote without actually inserting an extension controller.
		7.1.17	"Handling HOME" was moved from the "Wii Remote" section. The fact that the "HOME Menu is displayed immediately if HOME is pressed on the Wii Remote" was redundant with the "HOME Menu" section, so it was deleted. Changed the reference target to "HOME Menu."
		7.2.2	Moved "Handling HOME [Required]" to the "All Controllers" section.
		7.2.4	Changed the text "when the pointer is not being used" to read "in scenes that do not use the Pointer".
		7.2.5	Added to the description that output from the speaker won't be correct if it is being used at the same time as the Wii Remote memory.
		7.2.8 7.2.9	Deleted the example message. Added a reference to CONT_03 of the Wii Message List.
		7.2.10 7.2.11	Added that CONT_04 of the *Wii Message List* should be displayed if rewriting to or rereading from the Wii Remote memory fails three times.
		7.2.14	Deleted the example message. Added a reference to CONT_05 in the *Wii Message List*.
		7.2.15	Deleted the example message. Added a reference to CONT_06 and CONT_07 in the *Wii Message List*.
		7.2.16	Deleted, because it was verified that there is no need to prohibit continuous play on the Wii Remote speaker.
		7.2.19	Deleted the example message. Added a reference to CONT_09 in the *Wii Message List*.
		7.2.20	Deleted the example message. Added a reference to CONT_10 in the *Wii Message List*.
		7.2.23	Deleted the example message. Added a reference to CONT_11 in the *Wii Message List*.
		7.2.24	Deleted this paragraph; "One Time Mode Synchronization" will only be listed in the Wii Terminology, as is the case with all other terms.

Version	Revision Date	Item(s) Affected	Description
1.04	2007/05/31	7.2.25	Changed the heading from "Independent Adjustment of Wii Remote Speaker Volume by Applications" to "Prohibition of Using the WPADSetSpeakerVolume function." Deleted the condition "If the application has its own independent option setting for the Wii Remote speaker volume."
		7.2.26 7.2.27	Added the following new sections: • "Settings for Automatic Disconnection of Wii Remote Communications [Recommended]." • "Handling When a Wii Remote is Connected to a Player Number not Used by the Application [Recommended]."
		7.3 7.3.1	The content of "7.3.1 Application Support for the Nintendo GameCube Controller [Information]" is a description that relates to all Nintendo GameCube Controller sections, so this item was deleted from this 7.3.1 and moved to 7.3.
		8.3	Changed the handling of the following errors from "Errors that applications must handle" to "Errors that applications are expected but not required to handle (for which a bug report to Nintendo is expected)." • NAND_RESULT_ALLOC_FAILED • NAND_RESULT_BUSY
		8.4	Deleted the example message. Added a reference to NAND_01 and NAND_09 in the *Wii Message List*.
		8.11	Deleted the example message. Added a reference to NAND_02, NAND_03, and NAND_10 in the *Wii Message List*. Deleted description for saving multiple files. Added the description "When there is insufficient memory available and an insufficient number of available inodes at the same time, the insufficient memory available message should be given higher priority for displaying." Added a note about referencing the section on NAND_RESULT_MAXBLOCKS and NAND_RESULT_MAXFILES.
		8.13	Changed the rank from [Recommended] to [Required]. Deleted the example message. Added a reference to NAND_07 in the *Wii Message List*.
		8.14	Changed the rank from [Recommended] to [Required]. Deleted the example message. Added a reference to NAND_08 in the *Wii Message List*.
		8.15	Made it clear that this is an optional item for an occasion when one wants to save data immediately before the shutdown process. Because an addition was made to "Wii Power Button [Required]" regarding shutdowns via hardware, a note about referring to that section was added.

Version	Revision Date	Item(s) Affected	Description
1.04	2007/05/31	8.18	Updated this paragraph so that it indicates: That save banner files are created at the end of the sequence that creates the minimum required files starting at the state in which there are no files in the home directory. and That initial startup is an example of a state in which there are no files.
		8.25 8.26	Added the following new items: • "Prohibition of Usage Restrictions on Save Data Due to Wii Console-Specific Information [Required]." • "Support for NAND_RESULT_MAXBLOCKS and NAND_RESULT_MAXFILES [Recommended]."
		8.27	Moved the explanation about precautions when creating save banners from "12 Icons and Banners" to 8.27.
		9 9.1	The content of "Nintendo GameCube Memory Card Compatible Application [Information]" is a description that relates to all sections of Nintendo GameCube Memory Card, so this item was deleted from 9.1 and moved to Chapter 9 description.
		9.5	Added a correspondence chart between the conditions for error display and the *Wii Message List* IDs.
		10	Changed the heading from "Reset" to "Reset and Shutdown."
		10.1	The descriptions of exceptions relating to fatal errors of the optical disc drive was moved to a new section "Resetting and Shutting Down in Response to Fatal Errors [Required]" and deleted from here.
		10.2	The descriptions of exceptions relating to fatal errors of the optical disc drive was moved to be the new section "Resetting and Shutting Down in Response to Fatal Errors [Required]" and deleted from here. Added a note about shutdown via hardware.
		10.3	Revised the description about reset operations, following a change to the "Restrictions on Operations for Resets."
		10.4	Changed the term "hot reset" to specifically refer to the "OSRebootSystem function."
		10.5	The content of this paragraph was redundant with "Wii Console RESET [Required]" and "Wii Power Button [Required], so it was deleted.
		10.6	The former heading, "Prohibition Against Using the Nintendo GameCube Controller Reset Command [Required]," was changed to "Restrictions on Operations for Resets." Deleted the description about the Nintendo GameCube Controller reset command. Deleted "on the Wii Remote" from the phrase "the HOME Button Menu on the Wii Remote" (in the Japanese version).

Version	Revision Date	Item(s) Affected	Description
1.04	2007/05/31	11.2	Added that the HOME Menu must be displayed immediately if HOME is pressed, and that a confirmation message must not be displayed.
		11.4	Added "Settings and Display for 16:9 Screen Ratio Display Supported Application" as a reference.
		11.5	Added the handling when a Wii Remote is connected to a player number not used by the application following the addition of the new section "Handling When a Wii Remote is Connected to a Player Number not Used by the Application [Recommended]." Changed the heading from the former "Using HOME Menu from All Connected Wii Remotes" to "Using the HOME Menu from All Paired Wii Remotes."
		11.6	Changed the level of importance from [Required] to [Recommended].
		11.8 11.8.1 11.8.2	Restructured the former paragraph "Restrictions on Rumble when the Application has its own Rumble Settings [Required]" as the following three items: • Restrictions on Rumble Feature • Restrictions Due to the Rumble Setting of the HOME Menu [Required] • Prohibition of Changing the Rumble Setting of the HOME Menu [Required]
		11.11	Added the conditions for not displaying HOME. Changed the level of importance from [Required] to [Recommended].
		11.13	Added a new paragraph "Sound Volume When Entering and Exiting the HOME Menu [Required]."
		12.1	Added that animations are required in order to make the Wii Menu look like an interface to a television channel.
		12.8	Moved [this paragraph] to Chapter "8 Wii System Memory", since it describes cautions for creating save banners.
		12.9	Added the statement that sound is required in order to make the Wii Menu feel like an interface to a television channel.
		12.10	Added the [Required] notation because it had previously been missing.
1.03	2007/02/07	8	Notification When the Number of inodes or the Amount of Free Memory Is Insufficient [Required].

Version	Revision Date	Item(s) Affected	Description
1.02	2007/02/01	All	Combined precautions from the following three documents: ▪ Icon and Banner Specifications ▪ Wii Save Data Creation Guidelines ▪ HOME Menu Implementation Specifications In the Japanese version, the term "Wii system memory" has been replaced and standardized to "Wii system NAND memory" (no changes to the user terminologies for messages). The English version will continue to use "Wii system memory." Standardized the reference target name for individual sections
		2	Moved Startup Disc Error Handling [Required] to Chapter 3. Replaced the contents in Integrating Banners, Comments, and Icons [Information] with a reference to related items, and changed the level of importance. Referencing and Changing Wii System Settings updated to Application of Wii System Settings [Required]. Combined the two sections "Display of Health and Safety Warning" and "Application Startup Display Message" and updated as Prohibition of Warning Display by Applications [Required]. Updated Program Distribution Method [Required] and Parental Controls [Information]. Added [North American, European, and Australian Versions] License Display of Licensee Titles [Required].
		3	Updated Wii Game Disc Error Handling [Required]. Updated Normal Operation after the Disc Drive Motor Is Shut Off From a Period of Inactivity [Required]. Moved "Startup Disc Error Handling" from Chapter 3.
		6	Updated Support for Progressive Display [Recommended], PAL 50 Support [Required], Settings and Display for Applications That Support 16:9 Screen Ratio Display [Required], and Checking the On-Screen Display During Application Startup [Required]. Updated Applying Screen Burn-in Reduction Setting [Required] and changed the level of importance. Added Changing Screen Burn-in Reduction Wait Time [Recommended], Restrictions for Non-Activation of Screen Burn-in Reduction [Required], and Support for 16:9 Screen Ratio [Recommended].
		7.1	Updated Display When Extension Controller is Necessary [Required], Explanatory Message for Wii Remote Strap Use [Required, and Check Before Switching to Progressive Display [Required]. Updated the heading "Prohibition on the Use of Origin Reset Command for the Analog Input of the Extension controller" as Do Not Use Origin Reset Command for Extension Controller Analog Input as Game Input [Required]. Added Handling Unsupported Extension Controllers [Required].

Version	Revision Date	Item(s) Affected	Description
1.02	2007/02/01	7.2	Updated Handling HOME [Required], Disabling Pointer for Applications Without Pointer Function [Required], Disabling Pointer for Applications with Pointer Function [Recommended], Prohibition of Simultaneous Use of Wii Remote Memory and Wii Remote Speaker [Required], Wii Remote Data Error Checking [Required], Message Display When Writing to Wii Remote Memory [Required], Message Display When Loading Wii Remote Memory [Recommended], Data Compatibility for the Same Application Using Wii Remote Memory [Required]. Updated the heading "Prohibition of Application Dependent on Wii Remote Memory Access Speed" as Prohibition of Operations Dependent on Wii Remote Memory Access Speed [Required] and updated the content. Updated the heading "Handling Damage to Wii Remote Memory" as Handling Data Corruption in Wii Remote Memory [Recommended] and updated the content. Updated Continuous Play on the Wii Remote Speaker [Required]. Updated the heading "Audio Cut off of Speaker due to Poor Signal Sensitivity" as Speaker Sound Quality under Poor Signal Conditions [Recommended] and updated contents. Updated Displaying a Message Notifying the User That Battery Charge Is Low [Recommended]. Added Application of Pause Function at Termination of Wii Remote Communication [Information], Specification of Simple Pairing Procedure [Required], Message Displayed During Simple Pairing [Required], Display Format for Simple Pairing [Required], Prohibition of Using the WPADSetSpeakerVolume Function [Required].
		7.3	Updated the contents and the level of importance of Nintendo GameCube WaveBird Wireless Controller Support [Information].
		8	Updated Restrictions on Wii System Memory [Required] and Handling Corrupted Wii System Memory [Required]. Added Prohibition of Creating Files or Directories Without Owner Read Permission [Required], Prohibition of Frequent Read Access [Required], Save Banner File Creation [Required], Precautions for Creating Save Data [Information], Save Data Comments [Required], Save Data Location Specification [Required], Prohibition of Using Reserved Names [Required], Precautions Regarding the nocopy Directory [Required], and Save Data Tampering Protection by the Wii Console [Information].
		10	Updated Wii Console RESET [Required], Wii Power Button [Required], and Resetting the Nintendo GameCube Controller Origin during Reset Operations [Required].
		11	Added the chapter "Home Menu."
		12	Added the chapter "Icons and Banners."

Version	Revision Date	Item(s) Affected	Description
1.01	2006/10/06	2	Deleted "Boot Caching."
		3	Added "Prohibition on Drive Specification Dependency."
		4	Updated "Sound Setting," and "Sound Output for Applications Supporting Multiple Sound Settings."
		5.6	Edited brightness formula used for photosensitivity.
		6	Added "Video Settings." Updated "Settings and Display for EURGB60 Supported Application" and "Reset Operation during Progressive Output." Updated "Test of Different TV Reception Formats" as "Maintaining Appropriate Video Output." Added "Checking the On-screen Display during Application Startup" and "Check Before Switching to Progressive Display." Updated "Applying the Raster Burn Mitigation Setting."
		7	Updated "Explanatory Message for Strap Use." Updated "Using Wireless LAN" as "Frequency Band Usage when Using a Wireless LAN." Updated "Support for Changes in the Wii Remote and Sensor Bar Distance" to "Using the Relative Distance between the Wii Remote and the Sensor Bar." Added "Restrictions on Contents of Data Written to Wii Remote Memory." Updated "Continuous Play on the Wii Remote Speaker." Updated "Handling Audio Data Transmissions Stopped by the Feature for Avoiding Audio Cutoff" as "Audio Cutoff of Speaker due to Poor Signal Sensitivity."
		8	Updated "Write Operation Display," "Data Unit for Users," "Checking the Amount of Free Memory and Number of Free inodes Using NANDCheck[Async]," "Notification When the Amount of Free Memory or Number of inodes is Insufficient," and "Handling Corrupted Wii System Memory."
		9	Added "Prohibition on Programs Depending on the Access Speed of Nintendo GameCube Memory Cards."
		10	Updated "Wii Console RESET," and "Wii Power Button." Changed the title heading "Reacquisition of Origin for GameCube Controller during Reset Operation" to "Reset the Nintendo GameCube Controller Origin during Reset Operations."
1.00	2006/09/21	2	Updated "Prohibition of Dependency on Device Specifications."
		7.1	Updated "Handling +Control Pad Input."

Version	Revision Date	Item(s) Affected	Description
1.00	2006/09/21	7.2	Updated "Support for Changes in the Wii Remote and Sensor Bar Distance." Changed "Limitations on Use of HOME" to "Handling HOME." Added "Prohibition of Simultaneous Use of Wii Remote Memory and the Wii Remote Speaker." Added "Data Compatibility for the Same Application using Wii Remote Memory," "Prohibition of Application Dependent on Wii Remote Memory Access Speed," "Handling Damage to Wii Remote Memory," and "Displaying a Message when Overwriting Wii Remote Memory." Updated "Handling Audio Data Transmissions Stopped by the Feature for Avoiding Audio Distortion." Changed "Specifying a Message Notifying the User that Battery Charge is Low" to "Displaying a Message Notifying the User that Battery Charge is Low." Changed "Specifying a Message Notifying Users that Wii Remote Transmissions Have Terminated" to "Displaying a Message Notifying Users that Wii Remote Transmissions Have Terminated."
		8	Updated "Notification When the Amount of Free Memory or Number of inodes is Insufficient," "Handling Corrupted Files," and "Handling Corrupted Wii System Memory."
		9	Changed "Support for Non-Japanese Spec GameCube Memory Cards" to "Errors Displayed when Memory Cards Cannot be Used."
0.98	2006/09/15	Overall	Standardized terminology to use "Wii system memory."
		2	Added Paragraph 2.18, "Parental Controls."
		3	Added that the feature for stopping the disc drive motor does not work under Revolution SDK 2.2 Patch. Deleted "Error Display when Disc Eject Button is Pressed."
		5	Updated text. Added figures, graphs, and tables. Added information on calculating screen brightness.
		6	Added "Applying the Raster Burn Mitigation Setting."
		7.1	Updated "Explanatory Message for Wii Remote Strap Use." Updated "Lower Limit Value of the Control Stick and Button Analog Input Value," "Upper Limit Value of the Control Stick and Button Analog Input Value," and "Analog Input Value Range of Control Stick and Button." Deleted "Vibration Enabled/Disabled Mode Switch Function." Updated "Do Not Use the Origin Reset Command for the Analog Input to an Extension Controller."

Version	Revision Date	Item(s) Affected	Description
0.98	2006/09/15	7.2	Updated "Limit Use of HOME" and "Disabling Pointer for Application Without Pointer Function." Deleted "A Caution about Data Loss for Wii Remote Memory." Changed the level of importance of "Message Display when Writing to Wii Remote Memory." Standardized terminology to use the word "speaker." Combined "Handling Audio Cutoff of Speakers Due to Plugging in the External Extension Controller" and "Handling Audio Cutoff of Speakers Due to Packet Loss" to form "Handling Audio Data Transmissions Stopped by the Feature for Avoiding Audio Cutoff." Added "Specifying a Message Notifying the User that Battery Charge is Low" and "Specifying a Message Notifying Users That Wii Remote Transmissions Have Terminated."
		8	Updated "Checking the Number of Free inodes and Amount of Free Memory Using NANDCheck[Async]" and "Notification When the Number of inodes or the Amount of Free Memory Is Insufficient." Changed the rank of "Handling Corrupted Files." Updated "Handling Corrupted Wii System Memory."
		9	Updated "Memory Card Menu in Application." Updated "Note for Using CARDFastOpen Function" and changed the title to "Limit on Accessible Memory Card Files."
		10	Updated "Wii Console RESET" and "Wii Power Button."
0.97	2006/09/07	2	Updated "Boot Caching".
		4.4	Added "Prohibition on the Modification of the Audio Playback Frequency within an Application.
		6	Added description of Wii console setting values when using a 16:9 aspect ratio.
		7	Added a note that the screen for describing how to wear the strap is currently being studied. Added the paragraph "Prohibition on the Use of the Origin Reset Command for the Analog Input of the Extension Controller". Updated notes regarding message displayed when reading or writing to Wii Remote memory. Added the paragraphs "Handling Audio Cutoff of the Speaker Due to Connection of an External Extension Controller" and "Handling Audio Cutoff of the Speaker Due to Packet Loss."
		8	Standardized terminology to use the term "Wii system memory" (old term: Wii console internal flash memory) Deleted "Free Space Check" (the old 8.4) Updated paragraphs 8.1, 8.2, 8.3, 8.4, 8.5, and 8.8. Added new paragraphs from 8.9 to 8.15.
		10	Updated the names of documents being referenced. Added paragraphs 10.2 and 10.6. Changed the title of paragraph 10.5.

Version	Revision Date	Item(s) Affected	Description
0.96	2006/08/24	2.17	Deleted the (tentative) appellation for Boot Caching.
		6.17	Added paragraph on Support for 4:3 Aspect Ratio.
		7.2.4	Added indication that the pointer halt functionality is not currently available in the KPAD library.
		7.2	Added paragraphs on Wii Remote memory (7.2.5 to 7.2.9) and on the speaker (7.2.10 and 7.2.11).
0.95	2006/08/09	2.17	Added paragraph on "Boot Caching (tentative)."
		4	Reorganized the entire chapter.
		7.1.3	Changed the title "Warning Message of Moving Controllers" to "Explanatory Message for Strap Use," and upgraded the level of importance to "Required."
		8.9	Added paragraph on "Data Unit for Users."
0.94	2006/07/21		Initial version.