

Wii Programming Guidelines

Version: 1.01a

**The contents in this document are highly
confidential and should be handled accordingly.**

Confidential

These coded instructions, statements, and computer programs contain proprietary information of Nintendo of America Inc. and/or Nintendo Company Ltd. and are protected by Federal copyright law. They may not be disclosed to third parties or copied or duplicated in any form, in whole or in part, without the prior written consent of Nintendo.

Table of Contents

| | | |
|------|---|----|
| 1 | Overview | 11 |
| 1.1 | Ranking System | 11 |
| 1.2 | Wii Programming Guidelines Note | 11 |
| 2 | All | 12 |
| 2.1 | Prohibition of Dependency on Device Specifications [Information] | 12 |
| 2.2 | Memory and Register Initialization [Required] | 12 |
| 2.3 | Startup Time Restriction [Recommended] | 12 |
| 2.4 | Startup Disc Error Handling [Required] | 12 |
| 2.5 | Compliance with Legal Rights Display [Required] | 13 |
| 2.6 | Library Version [Recommended] | 13 |
| 2.7 | Deletion of Debug Information [Required] | 13 |
| 2.8 | Device Access without the Use of a Library Prohibited [Required] | 13 |
| 2.9 | Do Not Include Development Support Libraries in Release Version [Required] | 13 |
| 2.10 | Demo Screen Loop Processing [Required] | 14 |
| 2.11 | Integrating Banners, Comments, and Icons [Required] | 14 |
| 2.12 | Referencing and Changing Wii System Settings [Information] | 14 |
| 2.13 | Prohibition of Warning Screen Display by Application [Required] | 14 |
| 2.14 | [N. American Version Only] Onscreen Warning Display for Applications Using Online Features [Required] | 15 |
| 2.15 | Program Distribution Method [Required] | 15 |
| 2.16 | Parental Controls [Information] | 16 |
| 3 | Wii Game Disc and Drive | 17 |
| 3.1 | Prohibition on Drive Specification Dependency [Required] | 17 |
| 3.2 | Wii Game Disc Error Handling [Required] | 17 |
| 3.3 | Infinite Loops During Multiple Data Streaming Prohibited [Required] | 17 |
| 3.4 | [Only for Multiple Disc Games] Independent Startup of All Discs [Required] | 18 |
| 3.5 | Normal Operation after the Disc Drive Motor Is Shut Off From a Period of Inactivity [Required] | 18 |
| 3.6 | Display for Disc Drive Motor Shut Off From a Period of Inactivity [Recommended] | 18 |
| 3.7 | Data Layout for Double Layer Disc [Recommended] | 18 |

| | | |
|------|--|----|
| 4 | Sound | 19 |
| 4.1 | Sound Setting [Information] | 19 |
| 4.2 | Applying Wii Console System Sound Settings [Required] | 19 |
| 4.3 | Sound Output for Applications Supporting Multiple Sound Settings [Required]..... | 19 |
| 4.4 | Do Not Modify the Audio Playback Frequency Within an Application [Required]..... | 19 |
| 5 | Photosensitivity..... | 20 |
| 5.1 | About Photosensitivity and These Guidelines [Information] | 20 |
| 5.2 | Restrictions on Flashing Images and Lights [Recommended] | 21 |
| 5.3 | Restrictions on Flashing Saturated Red Colors [Recommended]..... | 21 |
| 5.4 | Restrictions on Image Reversals [Recommended] | 22 |
| 5.5 | Restrictions on Regular Patterns [Recommended] | 22 |
| 5.6 | Screen Brightness Calculations [Information] | 23 |
| 6 | Video..... | 26 |
| 6.1 | Display within the Safe Frame [Recommended] | 26 |
| 6.2 | Video Settings [Information] | 26 |
| 6.3 | Support for Progressive Display [Recommended] | 26 |
| 6.4 | Settings and Display for Progressive Display Supported Application [Required]..... | 26 |
| 6.5 | Switching Between Interlaced and Progressive Display Output [Recommended]..... | 27 |
| 6.6 | Addition of Interlaced/Progressive Switch Operation [Recommended] | 27 |
| 6.7 | Reset Operation during Progressive Output [Recommended]..... | 27 |
| 6.8 | Progressive Output Supported TV [Recommended] | 27 |
| 6.9 | PAL 50 Support [Required] | 27 |
| 6.10 | Settings and Display for EU RGB 60 Supported Application [Required]..... | 28 |
| 6.11 | Switching between PAL and EU RGB 60 [Recommended]..... | 28 |
| 6.12 | Addition of PAL/EU RGB 60 Switch Operation [Recommended] | 28 |
| 6.13 | Reset Operation during EU RGB 60 Output [Recommended] | 28 |
| 6.14 | Maintaining Appropriate Video Output [Required] | 28 |
| 6.15 | Display Method for 16:9 Screen Ratio [Recommended] | 29 |
| 6.16 | Settings and Display for Applications That Support 16:9 Screen Ratio Display [Required]..... | 29 |
| 6.17 | Trap Filter [Recommended] | 29 |
| 6.18 | Support for 4:3 Aspect Ratio [Required]..... | 29 |
| 6.19 | Checking the On-screen Display during Application Startup [Required] | 29 |
| 6.20 | Check Before Switching to Progressive Display [Required]..... | 30 |
| 6.21 | Applying the Raster Burn Mitigation Setting [Information]..... | 30 |

| | | |
|--------|--|----|
| 7 | Controllers..... | 31 |
| 7.1 | All Controllers..... | 31 |
| 7.1.1 | Display When Extension Controller is Necessary [Required] | 31 |
| 7.1.2 | Avoid Game Design Forcing Extended Period of Continuous Operation or Strenuous Activity [Information] | 31 |
| 7.1.3 | Explanatory Message for Wii Remote Strap Use (Required)..... | 32 |
| 7.1.4 | Handling +Control Pad Input [Required] | 32 |
| 7.1.5 | Operations on the Unused Buttons [Required] | 32 |
| 7.1.6 | Lower Limit Value of the Controller Stick and Button Analog Input Value [Recommended] | 33 |
| 7.1.7 | Upper Limit Value of the Controller Stick and Button Analog Input Value [Required] | 33 |
| 7.1.8 | Analog Input Value Range of Controller Stick and Button [Required] | 33 |
| 7.1.9 | Regarding Applications that Support the Rumble Feature [Required] | 34 |
| 7.1.10 | Avoid Continuous Operation of the Rumble Motor [Recommended] | 34 |
| 7.1.11 | Stop Rumble Motor When Controller Not in Use [Required]..... | 34 |
| 7.1.12 | Rumble Motor Run/Stop Switch Frequency Requirement [Recommended] | 34 |
| 7.1.13 | Hot Swapping Extension Controllers [Recommended] | 34 |
| 7.1.14 | Frequency Band Usage when Using a Wireless LAN [Information] | 34 |
| 7.1.15 | Do Not Use Origin Reset Command for Analog Input to Extension Controller [Required] | 35 |
| 7.2 | Wii Remote | 35 |
| 7.2.1 | Using the Relative Distance between the Wii Remote and the Sensor Bar [Recommended]..... | 35 |
| 7.2.2 | Handling HOME [Required]..... | 35 |
| 7.2.3 | Disabling Pointer for Applications Without Pointer Function [Required] | 35 |
| 7.2.4 | Disabling Pointer for Applications With Pointer Function [Recommended] | 35 |
| 7.2.5 | Do Not Use of Wii Remote Memory and Wii Remote Speaker Simultaneously [Required]..... | 35 |
| 7.2.6 | Restriction on Contents of Data Written to Wii Remote Memory [Required] | 36 |
| 7.2.7 | Checking for Errors in Wii Remote Data [Required]..... | 36 |
| 7.2.8 | Message Display When Writing to Wii Remote Memory [Required]..... | 36 |
| 7.2.9 | Message Display When Loading Wii Remote Memory [Recommended] | 36 |
| 7.2.10 | Error Handling When Writing to Wii Remote Memory [Required] | 36 |
| 7.2.11 | Error Handling When Reading From Wii Remote Memory [Required] | 36 |
| 7.2.12 | Data Compatibility for the Same Application Using Wii Remote Memory [Required] | 37 |
| 7.2.13 | Prohibition of Application Dependent on Wii Remote Memory Access Speed [Required] | 37 |
| 7.2.14 | Handling Damage to Wii Remote Memory [Recommended] | 37 |
| 7.2.15 | Displaying a Message when Overwriting Wii Remote Memory [Required]..... | 37 |
| 7.2.16 | Continuous Play on the Wii Remote Speaker [Required] | 37 |
| 7.2.17 | Halting Data Transfers When No Output Occurs on the Wii Remote Speaker [Recommended]..... | 37 |
| 7.2.18 | Audio Cutoff of Speaker due to Poor Signal Sensitivity [Recommended]..... | 37 |
| 7.2.19 | Displaying a Message Notifying the User That Battery Charge Is Low [Recommended] | 38 |
| 7.2.20 | Displaying a Message Notifying Users That Wii Remote Transmissions Have Terminated [Recommended]..... | 38 |

| | | |
|-------|--|----|
| 7.3 | Nintendo GameCube™ Controller | 38 |
| 7.3.1 | Application Support for the Nintendo GameCube Controller [Information] | 38 |
| 7.3.2 | Use of Origin Reset Command as Game Input Prohibited [Required] | 38 |
| 7.3.3 | Support for Origin Reset Command [Required] | 38 |
| 7.3.4 | Hot Swapping Support for Nintendo GameCube Controller Socket Devices [Required] | 39 |
| 7.3.5 | Frequency of Checks for Nintendo GameCube Controller Hot Swapping [Recommended] | 39 |
| 7.3.6 | Nintendo GameCube WaveBird Wireless Controller Support [Required] | 39 |
| 8 | Wii System Memory | 40 |
| 8.1 | Restrictions on Wii System Memory [Required] | 40 |
| 8.2 | Limits on Frequent Rewrites [Required] | 40 |
| 8.3 | Error Handling [Required] | 41 |
| 8.4 | Write Operation Display [Required] | 41 |
| 8.5 | Data Compatibility Within the Same Program [Required] | 42 |
| 8.6 | Applications Dependent on Wii System Memory Access Speed Prohibited [Required] | 42 |
| 8.7 | Reset Handling [Required] | 42 |
| 8.8 | Data Unit for Users [Required] | 42 |
| 8.9 | Checking the Number of Free i-nodes and Amount of Free Memory Using NANDCheck[Async] [Required] | 42 |
| 8.10 | Timing for Checking the Number of Free i-nodes and the Amount of Free Memory [Required] | 43 |
| 8.11 | Notification When the Number of i-nodes or the Amount of Free Memory Is Insufficient [Required] | 43 |
| 8.12 | Do Not Change Created File Size [Required] | 44 |
| 8.13 | Handling Corrupted Files [Recommended] | 44 |
| 8.14 | Handling Corrupted Wii System Memory [Recommended] | 44 |
| 8.15 | Saving Data During the Shutdown Process [Recommended] | 44 |
| 9 | Nintendo GameCube Memory Card | 45 |
| 9.1 | Nintendo GameCube Memory Card Compatible Application [Information] | 45 |
| 9.2 | Use of Write Functions Prohibited [Required] | 45 |
| 9.3 | Programs Must Not Depend on the Access Speed of Nintendo GameCube Memory Cards [Required] | 45 |
| 9.4 | Support for High Capacity Nintendo GameCube Memory Card [Required] | 45 |
| 9.5 | Errors Displayed when Memory Cards Cannot be Used [Required] | 46 |
| 9.6 | Compatible Slots [Recommended] | 46 |
| 9.7 | Memory Card Menu in Application [Required] | 46 |
| 9.8 | Limit on Accessible Memory Card Files [Required] | 46 |

| | | |
|------|--|----|
| 10 | Reset..... | 47 |
| 10.1 | Wii Console RESET [Required] | 47 |
| 10.2 | Wii Power Button [Required]..... | 47 |
| 10.3 | Reset the Nintendo GameCube Controller Origin during Reset Operations [Required] | 48 |
| 10.4 | Use of Hot Resets [Recommended] | 48 |
| 10.5 | Required Preparations Before Using the Reset/Shutdown Functions [Required] | 48 |
| 10.6 | Do Not Use the Nintendo GameCube Controller Reset Command [Required]..... | 48 |

Tables

| | | |
|-----------|--------------------------------|----|
| Table 2-1 | Rating Agencies by Market..... | 16 |
| Table 8-1 | Directory Size Limits..... | 40 |

Revision History

| Version | Revision Date | Description |
|---------|---------------|---|
| 1.01 | 10/6/2006 | <ul style="list-style-type: none"> Deleted "Boot Caching." Added "Prohibition on Drive Specification Dependency." Updated "Sound Setting," and "Sound Output for Applications Supporting Multiple Sound Settings." Added "Video Settings." Updated "Settings and Display for EU RGB 60 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." 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." Updated "Write Operation Display," "Data Unit for Users," "Checking the Amount of Free Memory and Number of Free i-Nodes Using NANDCheck[Async]," "Notification When the Amount of Free Memory or Number of i-Nodes is Insufficient," and "Handling Corrupted Wii System Memory." Added "Prohibition on Programs Depending on the Access Speed of Nintendo GameCube Memory Cards." Updated "Wii Console RESET Button," 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 | 9/21/2006 | <ul style="list-style-type: none"> Updated "Prohibition of Dependency on Device Specifications." Updated "Handling +Control Pad Input." Updated "Support for Changes in the Wii Remote and Sensor Bar Distance." Changed "Limitations on Use of HOME" to "Handling the 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 |

| | | |
|------|------------|---|
| | | <p>Charge is Low" to "Displaying a Message Notifying the User that Battery Charge is Low."</p> <ul style="list-style-type: none"> • Changed "Specifying a Message Notifying Users that Wii Remote Transmissions Have Terminated" to "Displaying a Message Notifying Users that Wii Remote Transmissions Have Terminated." • Updated "Notification When the Amount of Free Memory or Number of i-Nodes is Insufficient," "Handling Corrupted Files," and "Handling Corrupted Wii System Memory." • Changed "Support for Non-Japanese Spec GameCube Memory Cards" to "Errors Displayed when Memory Cards Cannot be Used." |
| 0.98 | 9/15/2006 | <ul style="list-style-type: none"> • Standardized terminology to use "Wii system memory." • Added Section 2.18, "Parental Control." • Added that the feature for stopping the disk drive motor does not work under Revolution SDK 2.2 Patch. • Deleted "Error Display when Disc Eject Button is Pressed." • Updated text. Added figures, graphs, and tables. Added information on calculating screen brightness. • Added "Applying the Raster Burn Mitigation Setting." • Updated "Explanatory Message for Wii Remote Strap Use." • Updated "Lower Limit Value of the Controller Stick and Button Analog Input Value," "Upper Limit Value of the Controller Stick and Button Analog Input Value," and "Analog Input Value Range of Controller 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." • 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 rank 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." • Updated "Checking the Number of Free i-nodes and Amount of Free Memory Using NANDCheck[Async]" and "Notification When the Number of i-nodes or the Amount of Free Memory Is Insufficient." • Changed the rank of "Handling Corrupted Files." • Updated "Handling Corrupted Wii System Memory." • Updated "Memory Card Menu in Application." • Updated "Note for Using CARDFastOpen Function" and changed the title to "Limit on Accessible Memory Card Files." • Updated "Wii Console RESET" and "Wii Power Button." |
| 0.97 | 09/07/2006 | <ul style="list-style-type: none"> • Updated "Boot Caching". • Added description of Wii console setting values when using a 16:9 aspect ratio. |

| | | |
|------|------------|--|
| | | <ul style="list-style-type: none">• Added a note that the screen for describing how to wear the strap is currently being studied.• Added the section "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.• 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 sections 8.1, 8.2, 8.3, 8.4, 8.5, and 8.8.• Added new sections from 8.9 to 8.15.• Updated the names of documents being referenced.• Added sections 10.2 and 10.6.• Changed the title of section 10.5. |
| 0.96 | 08/24/2006 | <ul style="list-style-type: none">• Deleted the (tentative) appellation for Boot Caching.• Added section on Support for 4:3 Aspect Ratio.• Added indication that the pointer halt functionality is not currently available in the KPAD library.• Added sections on Wii Remote memory (7.2.5 to 7.2.9) and on the speaker (7.2.10 and 7.2.11). |
| 0.95 | 08/09/2006 | <ul style="list-style-type: none">• Added section on "Boot Caching (tentative)."• Reorganized the entire section.• Changed the title "Warning Message of Moving Controllers" to "Explanatory Message for Strap Use," and upgraded the ranking to "Required."• Added section on "Data Unit for Users." |
| 0.94 | 07/21/2006 | <ul style="list-style-type: none">• Initial version. |

1 Overview

The Wii Programming Guidelines provide programming cautions for the Wii console and peripherals.

1.1 Ranking System

The following rankings indicate the level of importance of the 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.

Also, some of the documents referenced in the Wii Programming Guidelines are currently being created or updated, so they may not include the referenced information at this time. Please be patient as the contents will be added as they become available.

2 All

2.1 Prohibition of Dependency on Device Specifications [Information]

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 system memory, SD Memory Card, and Nintendo GameCube 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 assembler.

The arena is cleared to zero using `OSInit`.

2.3 Startup Time Restriction [Recommended]

Display screen images within 2 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 display mode to either progressive display or EU RGB 60 display.

Refer to the contents to be added to “Wii File Restriction” in “Revolution Optical Disc Drive Library (DVD)” for details.

2.4 Startup Disc Error Handling [Required]

Always include the disc error handling routine (including error display) in a program read by IPL.

If there is no error handling routine (error display) in the section read by the IPL, errors that occur before the reading of the error handling routine will not be processed.

Refer to “Revolution Optical Disc Drive Library (DVD)” for details on error handling.

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, refer to the instructions for the individual library tools. We also plan to distribute a document compiling various library tool rights displays.

2.6 Library Version [Recommended]

Use the newest library version available since libraries may be modified to fix problems.

2.7 Deletion of Debug Information [Required]

When compiling and assembling the release version, set options so that debug information will not be included in the program.

If debug information is included, it will not only decrease the execution speed, it will also increase the program size. Leaving in the debug code outputs may also result in lock-ups.

Link to the release versions of libraries instead of debug versions. When using `make`, the OS debug code is disabled by setting `NDEBUG=TRUE`.

2.8 Device Access without the Use of a Library Prohibited [Required]

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

The individual device libraries in the OS are created to avoid device compatibility issues and hardware bugs. Also, the newly created I/O devices are designed with library access in mind.

Accessing the individual devices without going through the appropriate library first may result in various unexpected problems.

2.9 Do Not Include Development Support Libraries in Release Version [Required]

Do not include development support libraries in the release version master data. Specifically, the `hio2` library falls into this category.

Including development support libraries may cause a Wii system malfunction.

2.10 Demo Screen Loop Processing [Required]

Make sure no problems occur when the demo screen (or the title screen, if there is no demo) is executed in a loop for more than 24 hours. Demo screens may loop overnight in store fronts.

Be especially careful of counter overflow. Also, refer to information about normal operation when the disc drive motor is shut off after a period of inactivity.

2.11 Integrating Banners, Comments, and Icons [Required]

Prepare a banner and comment for your Wii Game Disc.

For applications to be run on the Wii system memory, make an icon available in addition to a banner and a comment.

The comments should not exceed the system menu display area.

This is for the display in the system menu. See individual chapters for details.

2.12 Referencing and Changing Wii System Settings [Information]

Apply the various settings configured on the Wii console to the application program as applicable. Also, if these settings can be changed from the application side, apply these changes on the Wii console.

This is to allow easier sharing of user-configured values between applications.

Refer to the guidelines for individual Wii console configuration items for details.

2.13 Prohibition of Warning Screen Display by Application [Required]

The Wii system will display the health and safety warning screen when the user presses the Wii console Power Button. Nintendo believes that all warnings content are covered by this warning screen. Therefore, do not display any other warning message on the application side.

2.14 [N. American Version Only] Onscreen Warning Display for Applications Using Online Features [Required]

Applications using online features are required to display the following message specified by the ESRB, the local ratings establishment body:

"Game Experience May Change During Online Play"

The ESRB regulation only requires that the message be displayed before the use of an online feature and that the ability to circumvent is not permissible. However, with user appeal in mind, we recommend that this message be displayed in the startup menu screen. The font size must be 20 points or larger and the text must be noticeable and easy to read. For example, the message may be displayed along with other display messages, but it must not be in the same color as the background or require scrolling to be visible.

If it is clear that this message does not need to be displayed or ESRB confirms that this message display is not necessary, this will be considered inapplicable once the developer makes the request not to display the message.

This message is to be displayed regardless of the user's age.

2.15 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 read in programs or perform an external transfer of programs from the application using an SD Card, a Nintendo GameCube Memory Card, a network, USB, Bluetooth, or other kind of interface.

Also, even if data is simply being read using these interfaces, always perform buffer overflow checks or data encryption so that programs will not be executed as a result of buffer overflow.

2.16 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 will restrict 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 “Master Editor,” scheduled for release, and the “Parental Controls Programming Manual.”

3 Wii Game Disc and Drive

3.1 Prohibition on Drive Specification Dependency [Required]

Ensure that programs do not run uncontrollably or freeze due to 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 contents to be added to the “Revolution Optical Disc Drive Library (DVD),” including the messages, unless there is a special reason not to do so.

Having each application handle Wii Game Disc errors differently will confuse users.

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

3.3 Infinite Loops During Multiple Data Streaming Prohibited [Required]

Avoid infinite looping of simultaneous multiple instances of data streaming.

Data streaming is defined as the method of reading data sequentially as it is used, as opposed to a burst transfer where necessary data is loaded all at once from the disc before it is used. One good example of data streaming is movie playback.

Similarly, if frequent non-sequential access is necessary over a long period of time, try to make the access sequential or, in any case, prevent an infinite loop.

Also, seek noise created during simultaneous playback can be reduced significantly by placing files that are played simultaneously as close to each other as possible.

When the drive enters an infinite loop while constantly moving the drive head at high frequency, this may adversely affect the drive if the user leaves the game in this state.

If the application violates this requirement, the following steps may be taken to prevent such an occurrence.

- Interleave the data file and the audio file into a single file and play back the audio portion.
- Interleave multiple files for data streaming into a single file so that they can be accessed sequentially.

Another method to prevent infinite loops is to stop data streaming after several loops.

3.4 [Only for Multiple Disc Games] 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 standalone, have available a program that displays a message such as “This is <game title> Game Disc 2. Please insert <game title> Game Disc 1 and restart power.”

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]

Make sure applications using Wii Game Discs do not crash or malfunction when the drive motor shuts off.

When there is no controller input for a set period of time (TBD), the drive motor is shut off to extend drive life. This requirement allows the user to resume the game after a long period of inactivity.

Note: In the Revolution SDK 2.2 Patch 2, there is no need to support automatic shutoff of the disc drive motor due to controller inactivity for a fixed period of time. This is because the feature for stopping the disc drive motor does not work. Be sure to support automatic shutoff in future updates of the Revolution SDK when that feature becomes available.

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

When the drive motor shuts off, applications using discs should display a specified message (currently being studied) until normal operations are restored.

Note: In the Revolution SDK 2.2 Patch 2, there is no need to support automatic shutoff of the disc drive motor due to controller inactivity for a fixed period of time. This is because the feature for stopping the disc drive motor does not work. Be sure to support automatic shutoff in future updates of the Revolution SDK when that feature becomes available.

3.7 Data Layout for Double Layer Disc [Recommended]

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

Switching between the first and second layer adds extra latency.

4 Sound

4.1 Sound Setting [Information]

In the Wii sound setting menu, three selection types are available: mono, stereo, and surround.

Use the function `SCGetSoundMode` to acquire the system sound setting.

4.2 Applying Wii Console System Sound Settings [Required]

If the application supports the sound setting specified on the Wii console, it must use this sound mode.

If the application does not support surround or mono sound output, it should treat these settings as stereo.

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

If the application does not have a menu where the sound settings can be specified, obtain the Wii system sound settings at application startup and apply these to the application sound settings.

Also, if the application does have a sound setting menu, use the system sound setting as the initial value.

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/meter²), 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 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

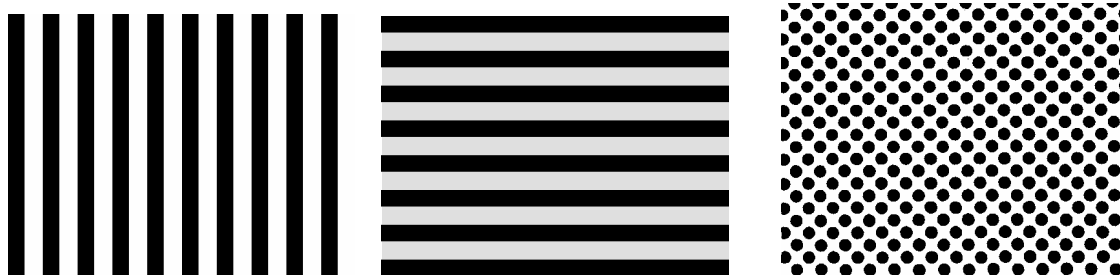
- (i) the luminance of the brighter element of the pattern is 30 nits or more and its contrast is greater than 40% (contrast is $(L1-L2)/(L1+L2)$, where $L1$ is the luminance of the brighter element of the pattern and $L2$ is the luminance of the darker), or
- (ii) the luminance of the brighter element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

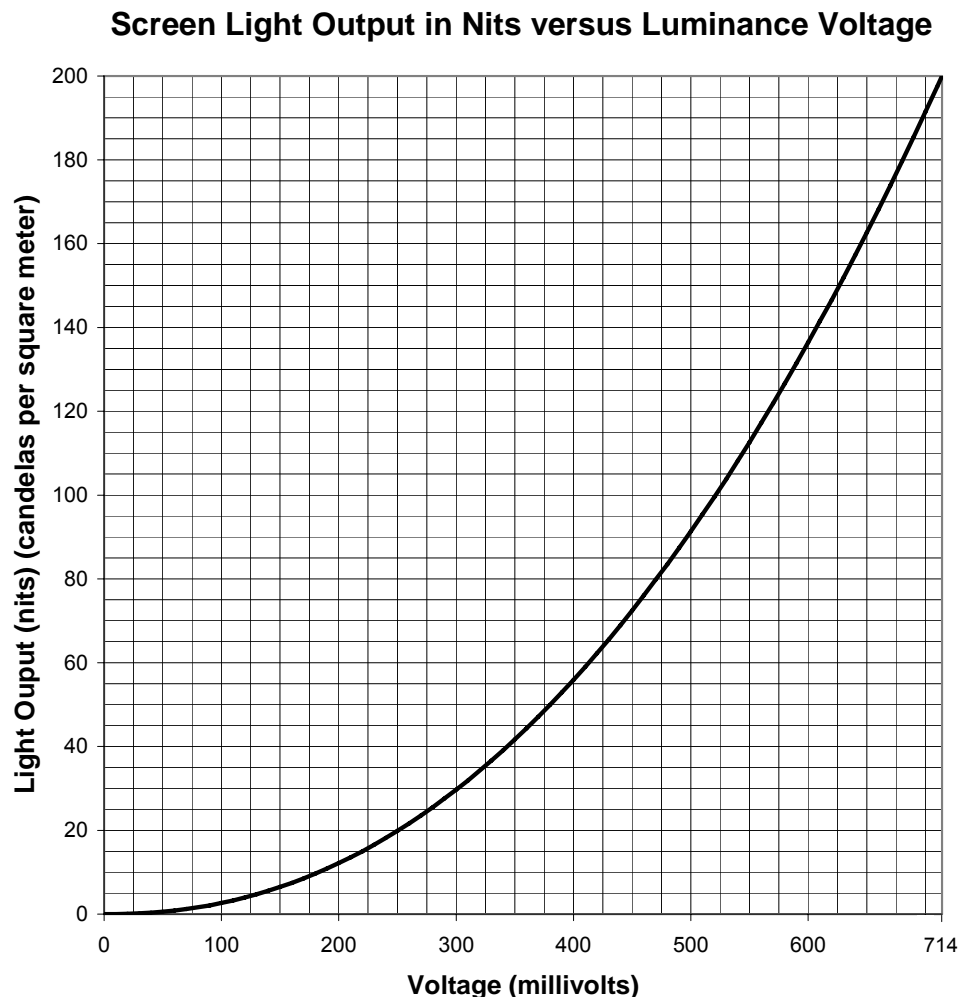
Checkerboard patterns and plaids are acceptable.



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.



| Voltage (mV) | Light (cd/m ²) |
|--------------|----------------------------|
| 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:

$$Y = 0.257 * R + 0.504 * G + 0.098 * B + 16$$

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:

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

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:

$$L = \left[\frac{213.618 * \left(\frac{R}{255} \right)^{\frac{1}{\gamma}} + 418.924 * \left(\frac{G}{255} \right)^{\frac{1}{\gamma}} + 81.458 * \left(\frac{B}{255} \right)^{\frac{1}{\gamma}}}{714} \right]^{2.2} * 200$$

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 will depend on the specific TV set. However, in general, the safe frame width will be approximately 84.4% of the maximum valid display width in the horizontal direction and 80.8% of the maximum valid display height in the vertical direction.

Refer to the “Revolution Video Interface Library” (VI.pdf) in the Revolution SDK for details on safe frames.

6.2 Video Settings [Information]

Wii console video settings include: an interlaced/progressive display setting, a 4:3/16:9 aspect ratio setting, a PAL/EU RGB 60 setting (Europe only), and an ON/OFF setting for the raster burn mitigation feature. If your applications support these options, the Wii console system settings must be referenced.

6.3 Support for Progressive Display [Recommended]

If the video format is NTSC, 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 for Progressive Display Supported Application [Required]

If the application supports progressive display, follow the display format specified on the Wii console to display in the appropriate format.

Applications not supporting the progressive display output may ignore the Wii console setting.

6.5 Switching Between Interlaced and Progressive Display Output [Recommended]

Perform display switching in the order: “black screen” → “switch display format” → “black screen” → “wait time of approximately 100 frames” → “screen with scenes or messages where progressive display is not very important”.

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

Refer to “Recommendations on Progressive Output” in the “Video Guidelines” for details on switch time.

6.6 Addition of Interlaced/Progressive Switch Operation [Recommended]

Please limit the number of switches between interlaced and progressive display to only once at application startup. Do not add other switch operations.

The screen display is distorted when switching between interlaced and progressive display. If the switch operation is performed frequently, it may take more than 100 frames (depending on the type of television set) before the screen returns to normal.

Refer to “Recommendations on Progressive Output” in the “Video Guidelines” for details on switch time.

6.7 Reset Operation during Progressive Output [Recommended]

When a reset is performed during progressive output, the current progressive output signal should be maintained and output resumed after the reset operation is complete.

Some TVs will fix the display ratio to 16:9 when a progressive signal is received. If the user is playing a game switched to the 4:3 ratio, the display ratio will switch to 16:9 when the progressive signal is interrupted.

6.8 Progressive Output Supported TV [Recommended]

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

Some TVs that support progressive output may perform scanline interpolation. You may not be able to achieve the desired screen display on such a TV.

6.9 PAL 50 Support [Required]

Always support PAL 50 in the PAL version of the application.

6.10 Settings and Display for EU RGB 60 Supported Application [Required]

If the application supports EU RGB 60, display in the appropriate format based on the mode configured on the Wii console.

Applications that do not support EU RGB 60 may ignore the Wii console setting.

6.11 Switching between PAL and EU RGB 60 [Recommended]

Perform display switching in the order: “black screen” → “switch display format” → “black screen” → “wait time of approximately 100 frames” → “screen with scenes or messages where EU RGB 60 display is not very important”.

During the switch between PAL and the EU RGB 60 display, the screen will be distorted and require few dozen frames before returning to normal.

Refer to “Recommendations on EU RGB 60” in the “Video Guidelines” for details on switch time.

6.12 Addition of PAL/EU RGB 60 Switch Operation [Recommended]

Please limit the number of switches between PAL and the EU RGB 60 display to the once at application startup. Do not add other switch operations.

The screen display is distorted when switching between PAL and EU RGB 60. If the switch operation is performed frequently, it may take more than 100 frames (depending on the type of television) before the screen returns to normal.

Refer to “Recommendations on EU RGB 60” in the “Video Guidelines” for details on switch time.

6.13 Reset Operation during EU RGB 60 Output [Recommended]

When a reset is performed during EU RGB 60 output, maintain the current EU RGB 60 output signal should be maintained and resumed after the reset operation is complete.

Some TVs will revert to 16:9 display ratio when EU RGB 60 signal is interrupted.

6.14 Maintaining Appropriate Video Output [Required]

The commercial version of the application must be tested for proper video output in the television reception format appropriate for the particular region (NTSC, PAL, MPAL). Please make sure the format does not change during a change in the game stage or after a reset operation.

Refer to “Regarding Reception Format” in the Video Guidelines.

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

When supporting a 16:9 screen ratio, we recommend the method that horizontally stretches the 4:3 ratio display.

This is because this method is the most common one as a TV function. Stretching an image from 4:3 to 16:9 also yields a higher resolution image compared to a letterbox display, which chops off the top and bottom of 4:3 ratio display.

When an aspect ratio of 16:9 has been set for the Wii console, the user is advised to switch the television to a wide-screen display by means of an on-screen message and a cautionary note in the Wii console user's 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 in the appropriate format according to the aspect ratio set for the Wii console.

Although it is all right for an application to have a separate aspect ratio setting, the default setting for the application should not only conform to the Wii console setting, but any changes to the application setting should not impact the Wii console setting.

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.

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 a game application can be displayed normally. Although game applications are not required to support all display modes (progressive, UERGB60, and 16:9), the Wii menu does include all display modes. For example, game applications that do not support progressive display must be displayed normally even if the Wii menu is being displayed in progressive mode.

6.20 Check Before Switching to Progressive Display [Required]

Check the following points before switching to progressive display.

- Check whether Wii D-Terminal cable or Wii Component Video cable is plugged into the AV Multi Out.
- Check whether the Wii console progressive setting is on.

For details on the actual confirmation method, see “Progressive” in the Video Programming Manual.

6.21 Applying the Raster Burn Mitigation Setting [Information]

Raster burn mitigation is a feature that decreases screen brightness to 25% if no controller operations have been detected for approximately five minutes. The user can adjust the Wii system setting to match his or her operating environment. Although applications should apply this setting if possible, this is not required when the playability of a game might be lost by doing so. This may occur in cases where there is a long movie animation lasting more than five minutes or when a game includes a mode that the user can enjoy viewing for extended periods of time without using the controller.

7 Controllers

7.1 All Controllers

7.1.1 Display When Extension Controller is Necessary [Required]

Display that a necessary extension controller is needed before its use in the application.

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

7.1.2 Avoid Game Design Forcing Extended Period of 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 stationary in a single location

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)

The purpose of this requirement is to reduce the possibility of user injuries or property damage from use of the Wii Remote and/or any controller connected to the Wii Remote.

When starting up an application using the Pointer, the Motion Sensor, or both, please 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.

The images and text are prepared in Japanese, English, German, French, Spanish, Italian, and Dutch respectively. For the U.S. and European versions of the application, which allow a user to set a language in the Wii hardware setting, display the language that properly reflects the hardware setting.

In case the game is intended for both the North American and European markets, be sure that the language setting made on the Wii console by the user is applied when displaying the reminder screen.

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 where you are holding it with both hands. Please contact Nintendo if you believe your game design does not require the wrist strap reminder screen.

7.1.4 Handling +Control Pad Input [Required]

Make sure that pressing the up and down or left and right directional buttons simultaneously 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:

- Attach priority levels to up and down/left and 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 the 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 in an application that use 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 Controller Stick and Button Analog Input Value [Recommended]

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

When a user is not touching the controller sticks or any buttons, the coordinates given by the controller 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 Controller Stick and Button Analog Input Value [Required]

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

Please keep the controller stick and button analog input value within the range set by Nintendo.

The resolution of individual controller 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 Controller Stick and Button [Required]

Verify proper operation with all controller 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 PAD library.

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 any 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 Avoid 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 life-span point of view.

7.1.11 Stop 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 Requirement [Recommended]

Do not switch between running and stopping the Rumble Motor at a frequency of more than 100 times a second.

Due to the wireless specifications, controlling the Rumble Motor at frequencies higher than this 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 Analog Input to Extension Controller [Required]

Do not use commands in applications which 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.

7.2 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]

When HOME is pressed, immediately display the Home Menu. (Please see the documents included with the Home Menu Library Package for guidelines on implementing the Home Menu.)

HOME is to be used only to display the Home Menu.

7.2.3 Disabling Pointer for Applications Without Pointer Function [Required]

Disable the Pointer in applications that do not use the Pointer functionality. This is to save battery power in the Wii Remote.

7.2.4 Disabling Pointer for Applications With Pointer Function [Recommended]

Even if the application does use the Pointer function, disable the Pointer when the Pointer is not being used. 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.

7.2.5 Do Not Use of Wii Remote Memory and Wii Remote Speaker Simultaneously [Required]

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

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 system memory when data is lost or to store temporary data that will not interfere with game progress even if lost.

7.2.7 Checking for Errors in Wii Remote Data [Required]

When using data from Wii Remote memory, be sure to employ one of the following methods to check for errors:

- Data duplication and error checking (recommended).
- Error correction capable of recovering from errors when contiguous 8-byte sections are lost.

However, there are no plans to provide error correction capable of recovering losses of contiguous 8-byte segments in future Revolution SDK releases.

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

Be sure to display a message such as “Do not connect to or disconnect from the extension controller” to prevent the user from inserting or removing the 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]

Be sure to display a message such as “Do not connect or disconnect the extension controller” 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 a verification after writing data to Wii Remote memory. If an error occurs during verification, attempt to re-write as many as three times.

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 re-read as many as three times, or use error correction.

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

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

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

The Wii Remote memory's access speed will depend on various external factors, including the radio status. Do not allow applications dependent on access speed to lock-up or stop as a result of this.

7.2.14 Handling Damage to Wii Remote Memory [Recommended]

If Wii Remote memory becomes corrupted, display a message to this effect to the user.

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

When overwriting data from another game in the Wii Remote memory, display one of the following messages:

- Data has been found from a game other than <game currently in play>. Is it OK to overwrite it?
- There is data from another game in memory. Is it OK to overwrite it?

7.2.16 Continuous Play on the Wii Remote Speaker [Required]

Do not play data that continues for eight minutes or more on the Wii Remote speaker.

For the specific method of use, see the "Wii Remote and Nunchuk Controllers Hardware Overview."

7.2.17 Halting Data Transfers When No Output Occurs on the Wii Remote Speaker [Recommended]

When there is no sound being played on the Wii Remote speaker (when the playing of such sound isn't needed), do not send data to the Wii Remote speaker.

We recommend this to prevent overuse of the Wii Remote battery.

There is also no need to frequently switch back and forth between enabling and disabling its use.

7.2.18 Audio Cutoff of Speaker due to Poor Signal Sensitivity [Recommended]

Audio cutoff of the Wii Remote Speaker sometimes occurs when radio signal conditions are poor. Be sure to program applications so that this audio cutoff does not stand out even when radio signal conditions deteriorate.

For the specific method to be used, see the "Wii Remote and Nunchuk Controller Hardware Overview."

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 such as, “The battery charge is running low.”

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]

Display a message such as, “Communications with the Wii Remote have been interrupted.”, to inform users when Wii Remote communications have been interrupted.

Although the user will know that the battery power has run out when the player indicator is turned off, the application might encounter problems if the battery in the Wii Remote were to suddenly become depleted.

Furthermore, if sudden loss of battery power is an issue, it might also be appropriate to pause the application in addition to displaying a message.

7.3 Nintendo GameCube™ Controller

7.3.1 Application Support for the Nintendo GameCube Controller [Information]

Applications do not need to support the Nintendo GameCube Controller. However, those applications that do support the Nintendo GameCube Controller must support the items discussed in paragraphs 7.3.2 through 7.3.5.

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

To avoid resetting the Control Stick, R Button, L Button, and Z Button 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 the Control Stick, R Button, L Button, and Z Button by holding down the X Button, Y Button, and START/PAUSE simultaneously, for 3 seconds.

7.3.3 Support for Origin Reset Command [Required]

Make sure the application will not react in an unexpected manner when the Control Stick and the R Button, L Button, and Z Button are reset.

This is to avoid user confusion.

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]

Make sure that there is no delay from the point at which the Controller is inserted until Controller operation becomes valid in the application. Otherwise, there may be disruption to the operation of the game.

7.3.6 Nintendo GameCube WaveBird Wireless Controller Support [Required]

For applications supporting the Nintendo GameCube Controller, please test for proper operation using WaveBird controllers in addition to standard Nintendo GameCube Controllers.

The internal operation of the API is different for the two controllers.

8 Wii System Memory

8.1 Restrictions on Wii System 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 Directory Size Limits

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

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

8.2 Limits on Frequent Rewrites [Required]

Please avoid unnecessary writing to the Wii system memory (such as writing to the Wii system memory by saving each second or every time a character moves) to prevent deterioration of that memory. If automatic saves are performed in the application, make sure to keep this below once a minute on average. If using a function in the `NANDSafe` series when automatically saving, save even less often (less than once every 6 minutes).

8.3 Error Handling [Required]

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

For details on error handling, refer to “NAND Processing Sequence” in the Revolution Function Reference manual.

The following table lists the error types used by the NAND library and how to handle them.

| Error Handling | Error |
|---|---|
| Error that applications must support | Response to NAND_RESULT_BUSY and NANDCheck[Async] |
| Errors that applications are expected to support, but support is not essential | NAND_RESULT_AUTHENTICATION NAND_RESULT_CORRUPT NAND_RESULT_ECC_CRIT NAND_RESULT_MAXBLOCKS NAND_RESULT_MAXFILES |
| Errors that must be eliminated during development or hidden from the end user's view | NAND_RESULT_ACCESS NAND_RESULT_EXISTS NAND_RESULT_INVALID NAND_RESULT_MAXFD NAND_RESULT_NOEXISTS NAND_RESULT_NOTEMPTY NAND_RESULT_OPENFD NAND_RESULT_FATAL_ERROR |
| Errors indicating there may be a problem with the library or errors that could be blamed on Nintendo (errors expected to be included in a bug report to Nintendo) | NAND_RESULT_ALLOC_FAILED NAND_RESULT_UNKNOWN |

8.4 Write Operation Display [Required]

When writing save data larger than 1 MB to the Wii system memory, please display a message or an icon indicating the data write operation is under way. This is to prevent the user from unintentionally destroying data by turning off power or resetting in the middle of a write operation to the Wii system memory.

For example, display “Continue saving. Please do not touch the Power Button or RESET.” before starting the write operation and wait for the user's input.

Also display “Saving in progress. Please do not touch the Power Button or RESET.” 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 things easier for the user to comprehend.

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 System Memory Access Speed Prohibited [Required]

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

There will be some variance between devices. Performance will also decrease as the device ages.

8.7 Reset Handling [Required]

If a reset is performed in the middle of a write operation, writing will be interrupted, causing a potential loss of data. Always perform a reset after verifying the end of the write operation.

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.

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 system memory file system.

8.9 Checking the Number of Free i-nodes and Amount of Free Memory Using NANDCheck[Async] [Required]

Before creating a new file or directory in the home directory, always check 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 i-nodes and enough free memory in the file system.

8.10 Timing for Checking the Number of Free i-nodes 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 i-nodes. 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 and gets the following message:

“Not enough memory. Cannot save.”

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 i-nodes or the Amount of Free Memory Is Insufficient [Required]

If the `NANDCheck[Async]` function finds that there are not enough available i-nodes or available memory, be sure to notify the user by displaying an appropriate message.

For example, display a message such as the following:

“There is not enough Wii system memory available. Free <X> block(s) of memory either by moving files to the SD card on the Data Management Screen or by deleting files.
[Start without saving]/[Return to Wii Menu Screen]”

In the case of applications that can save multiple files, either display the amount of memory required to save all files or display the amount of memory required for each individual file (or group of files). If information is displayed file-by-file, be sure to display a message like the following.

“There is not enough Wii system memory available. <X> block(s) of available memory are required to save <File Type 1>. <X> block(s) of available memory are required to save <File Type 2> ...”

When checking available resources at startup, it is all right to exclude those resources associated with files that are required only under certain conditions or in a given game mode as well as files for which saving is optional. In the case of files where available resources were not checked at startup, check available resources as they become required. Pay careful attention to the timing at which resources are checked so that users do not lose important data.

Anticipating that users may want to organize the contents stored in the Wii system memory themselves, the function `OSReturnToMenu` resets the system and brings up a screen used to edit the contents of the Wii system memory. Consider implementation of specifications where the user is notified when there are not enough resources, but where he or she can go to the screen for editing Wii system memory. If this type of processing is not embedded in software, display the following type of message on the screen.

“To edit the contents of Wii system memory, use the Data Management Screen on the Wii console.”

For details on displayed messages and available options, see the Wii Message List.

8.12 Do Not Change Created File Size [Required]

In general, do not change the size of a file once it has been created. If you do, the application may not be able to proceed due to insufficient memory caused by the size of a file having been increased while the application is running. Be sure to create files that are large enough to begin with.

Please contact Nintendo in the case of applications that are impractical unless the file size is variable.

8.13 Handling Corrupted Files [Recommended]

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 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.

See the Wii Message List for messages to display and options to present to the user.

8.14 Handling Corrupted Wii System Memory [Recommended]

Be sure to display a message like the following if `NAND_RESULT_CORRUPT` is returned as the result code for a NAND function.

“The Wii system memory has been corrupted. Refer to the Wii Operations Manual for further instructions.”

8.15 Saving Data During the Shutdown Process [Recommended]

If it is detected that the Power Button has been pressed, applications must quickly begin shutdown processing. If application data needs to be saved in Wii system memory during shutdown, make sure that saving can be completed within one second.

9 Nintendo GameCube Memory Card

9.1 Nintendo GameCube Memory Card Compatible Application [Information]

A Wii application does not need to be compatible with the Nintendo GameCube Memory Card. Applications that do support Nintendo GameCube Memory Card need to conform to paragraphs 9.2 through 9.7, below.

9.2 Use of Write Functions Prohibited [Required]

As a rule, use only the Read functions. While we do provide Write functions for debugging, this must not be used in the actual application.

Nintendo GameCube Memory Cards are available purely for read-only purposes, such as the transfer of previous Nintendo GameCube game data.

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

Ensure that programs do not run out of control or freeze due to a dependence on the access rate of Nintendo GameCube Memory Card. This requirement is due to the fact that there is individual variation among devices. In addition, performance will drop over time due to degradation.

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

Make sure the application will operate normally when a high-capacity Nintendo GameCube Memory Card (maximum of 1019 blocks) is inserted.

High-capacity Memory Cards may have different capacities (total block number) or access speeds.

For example, if the application fixes the Memory Card capacity to 59 blocks, the program may freeze or display incorrectly due to missing block number digits.

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

If an application attempts to use a memory card in one of the following states, display an appropriate message to notify the user that the card cannot be used:

- A device other than a Memory Card has been inserted.
- The Memory Card is physically damaged.
- The data on the Memory Card's system blocks is corrupted.
- The card is formatted for another market.
- The data was corrupted while reading save data.

9.6 Compatible Slots [Recommended]

The Wii console comes with two Nintendo GameCube Memory Card Slots: Slot A and Slot B. Allow operation from both Memory Card slots, if possible. If only one slot will be supported, make sure to display a message indicating which slot should be used.

If only one slot is supported without any informational message, users will not be able to tell which slot to use for the application Memory Card.

9.7 Memory Card Menu in Application [Required]

Handle only those files used by an application in the Memory Card menu inside the application. List only the files used by the application even when listing files. Do not create a Memory Card menu that is capable of displaying and manipulating all files as is done on the system menu Memory Card screen.

This is because a special operation is necessary to fulfill specifications equivalent to the Wii system menu Memory Card screen.

9.8 Limit on Accessible Memory Card Files [Required]

When an application accesses files on a memory card, allow access only to files created by or used with your own applications intended for the Wii system. Do not access other files.

10 Reset

10.1 Wii Console RESET [Required]

If RESET is pressed while an application is executing, always perform a reset operation immediately once the button is pressed. Also, do not perform any operation other than a reset when RESET is used.

As an exception to the above, we recommend that RESET be disabled during the display of optical disc drive 'fatal error' messages and the NAND_RESULT_CORRUPT error used with Wii system memory. (In other words, give priority to 'fatal error' messages and the NAND_RESULT_CORRUPT error.)

This is to avoid user confusion.

Refer to the Wii Console Reset/Shutdown Programming Manual ([Reset_Shutdown.pdf](#)) and Revolution Optical Disc Drive Library ([DVD.pdf](#)), in the Revolution SDK, for details.

10.2 Wii Power Button [Required]

If the Power Button on the Wii console or Wii Remote is pressed while an application is executing, always be sure to perform a shutdown immediately once the Power Button is pressed. Also, do not perform any operation other than a shutdown when the Power Button is pressed.

As an exception to the above, we recommend that the Power Button be disabled during the display of fatal error messages for the optical disc drive and the NAND_RESULT_CORRUPT error used with Wii system memory. (In other words, give priority to fatal error messages and the NAND_RESULT_CORRUPT error.)

This is to avoid user confusion.

Refer to the Wii Console Reset/Shutdown Programming Manual ([Reset_Shutdown.pdf](#)) and Revolution Optical Disc Drive Library ([DVD.pdf](#)), in the Revolution SDK, for details.

10.3 Reset 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 using the Controller or the Wii console RESET.

If the function `OSResetSystem` is used during the reset operation, an origin reset will be performed inside the function. However, if `OSResetSystem` is not used for the reset operation, call the function `PADRecalibrate` to reacquire the origin.

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

10.4 Use of Hot Resets [Recommended]

Avoid the use of hot resets as much as possible.

It takes more time for an application to restart when a hot reset is used because the boot screen is displayed.

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

Close all running processes before executing the reset or shutdown function. Otherwise, reset operations may fail or not run properly.

Refer to the Wii Console Reset/Shutdown Programming Manual for details.

10.6 Do Not Use the Nintendo GameCube Controller Reset Command [Required]

The act of simultaneously pressing the B Button, the X Button, and START/PAUSE was designed to reset Nintendo GameCube applications from the Controller. Although Wii applications may support the use of Nintendo GameCube Controllers, they must not accept the above command for the purpose of resetting an executing application.

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.

© 2006 Nintendo

The contents of this document cannot be duplicated, copied, reprinted, transferred, distributed or loaned in whole or in part without the prior approval of Nintendo.