EWF Management Software
Windows driver software for Classemby Devices®/Industrial Controller

Help for Windows
# Contents

<table>
<thead>
<tr>
<th>Chapter 1  Introduction</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Overview ......................................................................</td>
<td>3</td>
</tr>
<tr>
<td>1.2 Features ......................................................................</td>
<td>3</td>
</tr>
<tr>
<td>Chapter 2  Product Specifications</td>
<td>4</td>
</tr>
<tr>
<td>2.1 Functional Specifications .......................................</td>
<td>4</td>
</tr>
<tr>
<td>2.2 Product Composition ...............................................</td>
<td>4</td>
</tr>
<tr>
<td>Chapter 3  Programming Guide</td>
<td>5</td>
</tr>
<tr>
<td>3.1 Installation ..................................................................</td>
<td>5</td>
</tr>
<tr>
<td>3.2 Programming Guide for Memory Monitoring Library ..........</td>
<td>5</td>
</tr>
<tr>
<td>3.2.1 Retrieving Disk Space that EWF Consumes ..................</td>
<td>5</td>
</tr>
<tr>
<td>3.2.2 Retrieving Physical Disk Space ................................</td>
<td>5</td>
</tr>
<tr>
<td>Chapter 4  Reference</td>
<td>6</td>
</tr>
<tr>
<td>4.1 List of DLL Function ...............................................</td>
<td>6</td>
</tr>
<tr>
<td>4.1.1 IfEwfSetEventEwfRamData .......................................</td>
<td>7</td>
</tr>
<tr>
<td>4.1.2 IfEwfGetEwfRamData ...............................................</td>
<td>9</td>
</tr>
<tr>
<td>4.1.3 IfEwfGetPhysicalMemoryEx .......................................</td>
<td>10</td>
</tr>
<tr>
<td>Chapter 5  Sample Program</td>
<td>11</td>
</tr>
<tr>
<td>5.1 Execution Procedure ................................................</td>
<td>11</td>
</tr>
<tr>
<td>5.2 List of Sample Program ............................................</td>
<td>11</td>
</tr>
<tr>
<td>Chapter 6  Interface EWF Manager</td>
<td>12</td>
</tr>
<tr>
<td>6.1 Interface EWF Manager .............................................</td>
<td>12</td>
</tr>
<tr>
<td>6.2 Starting Interface EWF Manager ..................................</td>
<td>12</td>
</tr>
<tr>
<td>6.3 Commands .................................................................</td>
<td>12</td>
</tr>
<tr>
<td>6.3.1 Setting in the Dialog Box ......................................</td>
<td>13</td>
</tr>
<tr>
<td>Chapter 7  Terms of Use</td>
<td>15</td>
</tr>
<tr>
<td>7.1 Limited Warranty ....................................................</td>
<td>15</td>
</tr>
<tr>
<td>7.2 Copyrights and Intellectual Property Rights ................</td>
<td>15</td>
</tr>
<tr>
<td>7.3 Warning Regarding Medical and Clinical Use of Our Products</td>
<td>15</td>
</tr>
<tr>
<td>7.4 Prohibition of Reproduction ......................................</td>
<td>15</td>
</tr>
<tr>
<td>7.5 Limitation of Liability ...........................................</td>
<td>15</td>
</tr>
<tr>
<td>7.6 Trademark ...............................................................</td>
<td>15</td>
</tr>
</tbody>
</table>

---

Copyright 2013 Interface Corporation. All rights reserved.
Chapter 1  Introduction

1.1 Overview

The EWF Management Software includes Interface EWF Manager and Memory Monitoring library. Interface EWF Manager configures EWF (Enhanced Write Filter) settings and monitors each memory with GUI (Graphical User Interface). Memory Monitoring library monitors the disk space that EWF consumes from your application software running on Windows. Application software links a provided DLL (dynamic link library) and monitors the disk space that EWF consumes through the API (application programming interface).

This document guides you to use EWF Management Software on Windows.

1.2 Features

- You can easily configure EWF with the GUI utility program.
- The disk space that EWF consumes and physical free disk space can be confirmed with the GUI utility program. When the values exceed the threshold value, a warning message will appear.
- This software retrieves the status of disk space that EWF consumes and physical disk space with the library.
- This software notifies an event when the disk space that EWF consumes exceeds the threshold value. A cycle to check whether the disk space is exceeds the threshold value can be specified in ms.
Chapter 2  Product Specifications

2.1 Functional Specifications

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface EWF Manager</td>
<td>- Enables/disables EWF every drive.</td>
</tr>
<tr>
<td></td>
<td>- Displays a warning message when the disk space exceeds specified threshold value.</td>
</tr>
<tr>
<td></td>
<td>- Monitors disk space that EWF consumes and physical free disk space.</td>
</tr>
<tr>
<td></td>
<td>- Enables/disables HORM.</td>
</tr>
<tr>
<td></td>
<td>- Executes COMMIT to the specified drive.</td>
</tr>
<tr>
<td>Memory Monitoring library</td>
<td>Disk space that EWF consumes.</td>
</tr>
<tr>
<td></td>
<td>- Retrieves disk space that EWF consumes.</td>
</tr>
<tr>
<td></td>
<td>- Specifies the threshold value of event notification in bytes.</td>
</tr>
<tr>
<td></td>
<td>- Specifies the monitoring cycle of event notification in microseconds (ms).</td>
</tr>
<tr>
<td>Physical disk space</td>
<td>Retrieves disk space and free disk space.</td>
</tr>
</tbody>
</table>

2.2 Product Composition

<table>
<thead>
<tr>
<th>Item</th>
<th>File Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latest information</td>
<td>README.HTM</td>
<td>Latest information</td>
</tr>
<tr>
<td>Installer</td>
<td>SETUP.EXE</td>
<td>Installation sample program</td>
</tr>
<tr>
<td>Sample program</td>
<td>volevent</td>
<td>Event notification program</td>
</tr>
<tr>
<td>Utility</td>
<td>IFEWF.EXE</td>
<td>Interface EWF Manager</td>
</tr>
<tr>
<td>DLL</td>
<td>IFEWF.DLL</td>
<td>Dynamic link library file</td>
</tr>
<tr>
<td></td>
<td>IFEWF.LIB</td>
<td>Import library file</td>
</tr>
<tr>
<td>Header files</td>
<td>IFEWF.H</td>
<td>Header file for Visual C++</td>
</tr>
<tr>
<td></td>
<td>IFEWF.BAS</td>
<td>Header file for Visual Basic</td>
</tr>
<tr>
<td>Help</td>
<td>HELP.PDF</td>
<td>Help (PDF file)</td>
</tr>
</tbody>
</table>
Chapter 3  Programming Guide

3.1 Installation
Refer to README.HTM for installation.

3.2 Programming Guide for Memory Monitoring Library
This section explains basic control procedures. The examples are written in C.

3.2.1 Retrieving Disk Space that EWF Consumes
The IfEwfGetEwfRamData function retrieves current disk space that EWF consumes in bytes.

```c
INT Ret;
LONGLONG RamSize;

// Retrieval of disk space that EWF consumes
Ret = IfEwfGetEwfRamData(&RamSize);
```

When using an event, specify a threshold value for disk space that EWF consumes, event handle that notifies you when disk space exceeds the threshold value, and cycle to monitor if disk space exceeds the threshold value.

```c
INT Ret;
HANDLE VolEvent;
LONGLONG RamDataBytes;

// Event generation
VolEvent = CreateEvent(NULL, TRUE, FALSE, NULL);

// Threshold value (256 MB), event handle, and cycle (5 s) are set
RamDataBytes = 256*1024*1024;
Ret = IfEwfSetEventEwfRamData(RamDataBytes, VolEvent, 5*1000);
```

When disk space that EWF consumes exceeds the specified threshold value, the specified signal event gets to signal status.

3.2.2 Retrieving Physical Disk Space
The IfEwfGetPhysicalMemory function retrieves a current physical disk space and free disk space.

```c
INT Ret;
ULONGLONG Total;
ULONGLONG Free;

Ret = IfEwfGetPhysicalMemoryEx(&Total, &Free);
```
### 4.1 List of DLL Function

<table>
<thead>
<tr>
<th>No.</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>IfEwfSetEventEwfRamData</code></td>
<td>Configures an event for disk space that EWF consumes.</td>
</tr>
<tr>
<td>2</td>
<td><code>IfEwfGetEwfRamData</code></td>
<td>Retrieves current disk space that EWF consumes.</td>
</tr>
<tr>
<td>3</td>
<td><code>IfEwfGetPhysicalMemoryEx</code></td>
<td>Retrieves the current information of physical disk space.</td>
</tr>
</tbody>
</table>

**<EWF memory>**

**<Physical memory>**
4.1.1 IfEwfSetEventEwfRamData

**Description**

The IfEwfSetEventEwfRamData function configures and clears an event for disk space that EWF uses.

**Syntax**

C

```c
INT IfEwfSetEventEwfRamData(
    LONGLONG RamDataBytes,
    HANDLE EventHandle,
    ULONG WatchCycle
);
```

Visual Basic

```vbnet
Declare Function IfEwfSetEventEwfRamData Lib "ifewf.dll"( 
    ByVal RamDataBytes As Currency, 
    ByVal EventHandle As Long, 
    ByVal WatchCycle As Long 
) As Long
```

**Parameters**

- **RamDataBytes**
  The parameter specifies a threshold value for disk space that EWF consumes in bytes. It switches an event specified in EventHandle to the signal state when the disk space that EWF consumes exceeds a specified threshold value. The Currency type in Visual Basic internally sets a value to one ten thousandth of actual value. Therefore, multiply the actual value by 10000 in Visual Basic. Refer to "Chapter 5  Sample Program " and “Example” for more details.

- **EventHandle**
  The parameter specifies an event handle to switch to the signal state when disk space that EWF consumes is greater than a threshold value specified with RamDataBytes. To invalidate a registered event, specify the parameter as follows:
  - C : NULL
  - Visual Basic: 0

- **WatchCycle**
  The parameter specifies a cycle to monitor if disk space that EWF consumes exceeds the threshold value. The unit is microseconds (ms).

**Return Values**

This software returns 0 if the function is successfully closed. If the function returns a value other than 0, check for an invalid input parameter and multiple registrations.

**Comment**

Overwriting this function will cause an error. To change the threshold value or monitoring cycle, specify the EventHandle parameter as follows and cancel the event.
- C: NULL
- Visual Basic: 0
Example
Setting the threshold value to 256 MB and the monitoring cycle to 5 s

C

```c
INT Ret;
HANDLE VolEvent;
LONGLONG RamDataBytes;

// Event generation
VolEvent = CreateEvent(NULL, TRUE, FALSE, NULL);

// Event generation
RamDataBytes = 256*1024*1024;
Ret = IfEwfSetEventEwfRamData(RamDataBytes, VolEvent, 5*1000);
```

Visual Basic

```vbnet
Dim Ret As Long
Dim VolEvent As Long
Dim RamDataBytes As Currency
Dim WaitRet As Long

' Event generation
VolEvent = CreateEvent(0, True, False, 0)

' Event generation
RamDataBytes = 256
RamDataBytes = RamDataBytes * 1024
RamDataBytes = RamDataBytes * 1024
RamDataBytes = RamDataBytes / 10000
Ret = IfEwfSetEventEwfRamData(RamDataBytes, VolEvent, 5 * 1000)
```
4.1.2 IfEwfGetEwfRamData

Description
The IfEwfGetEwfRamData function retrieves current disk space that EWF consumes.

Syntax
C
INT IfEwfGetEwfRamData(
  LONGLONG* RamDataBytes
);

Visual Basic
Declare Function IfEwfGetEwfRamData Lib "ifewf.dll"( _
  ByRef RamDataBytes As Currency _
) As Long

Parameters
RamDataBytes Storage destination for current disk space that EWF consumes. The unit is bytes. The Currency type in Visual Basic internally sets a value to one ten thousandth of actual value. Therefore, multiply the actual value by 10000 in Visual Basic. Refer to “Example” for details.

Return Value
This software returns 0 if the function is successfully closed. If the function returns a value other than 0, check for the input parameter.

Example
Retrieving current disk space that EWF consumes

C
INT Ret;
LONGLONG RamSize;
Ret = IfEwfGetEwfRamData(&RamSize);

Visual Basic
Dim Ret As Long
Dim RamSize As Currency
Ret = IfEwfGetEwfRamData(RamSize)
RamSize = RamSize * 10000
4.1.3 IfEwfGetPhysicalMemoryEx
The IfEwfGetPhysicalMemoryEx function retrieves the information of physical disk space.

Syntax
C

```c
INT IfEwfGetPhysicalMemoryEx(
    ULONGLONG* TotalBytes,
    ULONGLONG* FreeBytes
);
```

Visual Basic

```vbnet
Declare Function IfEwfGetPhysicalMemoryEx Lib "ifewf.dll"(_
    ByRef TotalBytes As Currency, _
    ByRef FreeBytes As Currency_
) As Long
```

Parameter

- **TotalBytes**
  - The parameter specifies a storage destination for physical disk space in bytes.
  - The Currency type in Visual Basic internally sets a value to one ten thousandth of actual value. Therefore, multiply the actual value by 10000 in Visual Basic. Refer to “Example” for details.

- **FreeBytes**
  - The parameter specifies a storage destination for free physical disk space in bytes.
  - The Currency type in Visual Basic internally sets a value to one ten thousandth of actual value. Therefore, multiply the actual value by 10000 in Visual Basic. Refer to “Example” for details.

Return Value
This software returns 0 if the function is successfully closed.
If the function returns a value other than 0, check for the input parameter.

Example
Retrieving physical disk space and free physical disk space

C

```c
INT Ret;
ULONGLONG Total;
ULONGLONG Free;

Ret = IfEwfGetPhysicalMemoryEx(&Total, &Free);
```

Visual Basic

```vbnet
Dim Ret As Long
Dim Total As Currency
Dim Free As Currency

Ret = IfEwfGetPhysicalMemoryEx(Total, Free)
Total = Total * 10000
Free = Free * 10000
```
Chapter 5  Sample Program

5.1 Execution Procedure

This product provides sample program files for C++ and Visual Basic. Executable files of the sample programs are NOT included with this product. Compile the source code and create executable file to start.

Visual C++
2. Select File > Open Workspace.
3. Open the makefile, *.dsp.
4. Build the project file.
5. Run the executable file, *.exe.

Visual Basic
2. Open the project file, *.vbp.
3. Build the project file.
4. Run the executable file, *.exe.

5.2 List of Sample Program

<table>
<thead>
<tr>
<th>Sample program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>volevent</td>
<td>Monitors the disk space that EWF consumes every 5 seconds to check if the disk space exceeds 256 MB. When it exceeds 256 MB, a message will appear and the program will be terminated.</td>
</tr>
</tbody>
</table>
Chapter 6  Interface EWF Manager

6.1 Interface EWF Manager

Interface EWF Manager supports the following functions.

- Enabling and disabling EWF per drive
- Executing COMMIT operation to a specific drive (only once)
- Monitoring disk space that EWF consumes, physical disk space, and free physical disk space
- Specifying thresholds of EWF consumed disk space and free physical disk space, and warning with a message when the thresholds are exceeded.
- Enabling and disabling HORM

6.2 Starting Interface EWF Manager

This utility program runs automatically when the operating system starts.

Check the task tray to confirm that the utility is running.
The system tray displays a maple leaf icon while the utility is running.

6.3 Commands

Right-clicking the icon of Interface EWF Manager opens the following command window.

The following table explains each command.

<table>
<thead>
<tr>
<th>Commands</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setting</td>
<td>Opens the dialog box described in &quot;6.3.1 Setting in the Dialog Box,” to configure EWF.</td>
</tr>
<tr>
<td>Help</td>
<td>Opens this Help file.</td>
</tr>
<tr>
<td>Information</td>
<td>Displays the version and our website URL.</td>
</tr>
<tr>
<td>Exit</td>
<td>Closes Interface EWF Manager. The icon in the system tray will disappear after this command is executed.</td>
</tr>
</tbody>
</table>
6.3.1 Setting in the Dialog Box

After you select the Setting command, the following dialog box appears.

The following table explains each element in the dialog box.

<table>
<thead>
<tr>
<th>Items</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HORM Enable</td>
<td>When you select the check box, HORM will be enabled and the operating system boots rapidly at the next boot.</td>
</tr>
<tr>
<td></td>
<td><strong>Follow instructions described below to enable HORM. Reboot is required to disable HORM.</strong></td>
</tr>
<tr>
<td></td>
<td>1. Disable EWF on drive C beforehand to save the setting changes.</td>
</tr>
<tr>
<td></td>
<td>2. Enable <strong>Hibernation</strong> from <strong>Power Option</strong> setting screen and reboot the operating system. (For Windows Embedded Standard 7, execute</td>
</tr>
<tr>
<td></td>
<td>“powercfg /h on” from the command prompt.)</td>
</tr>
<tr>
<td></td>
<td>3. Enable EWF on all drives with Interface EWF Manager, then reboot the operating system.</td>
</tr>
<tr>
<td></td>
<td>4. Enable HORM and make the computer hibernated.</td>
</tr>
<tr>
<td></td>
<td>5. The industrial controller will boot rapidly.</td>
</tr>
<tr>
<td>EWF C to Z</td>
<td>If you select check boxes from C through Z, EWF on the selected drives will be enabled. The change will be reflected after rebooting the operating system.</td>
</tr>
<tr>
<td></td>
<td>* The item is displayed only when target drives for EWF exist. Even if drives exist, it will not be displayed unless they are target for EWF.</td>
</tr>
<tr>
<td></td>
<td>Click the <strong>All</strong> button to enable EWF on all the drives listed here. (When all check boxes of drives are selected, click the <strong>All</strong> button to disable EWF on all the drives.)</td>
</tr>
<tr>
<td>Items</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Warning EWF Threshold</td>
<td>Threshold value of disk space that EWF consumes. The unit is KB. When EWF consumes larger disk space than the threshold value specified here, a warning message will appear. If you clear the check box, the warning message will not appear.</td>
</tr>
<tr>
<td>EWF RAM</td>
<td>Current disk space that EWF consumes. The unit is KB. (Write-protected)</td>
</tr>
<tr>
<td>Free Threshold</td>
<td>Threshold value of physical disk space. The unit is KB. When the physical disk space becomes smaller than the specified threshold value, a warning message will appear. If you clear the check box, the warning message will not appear.</td>
</tr>
<tr>
<td>Free Memory</td>
<td>Current physical free disk space. The unit is KB. (Write-protected)</td>
</tr>
<tr>
<td>Commit Total Memory</td>
<td>Physical disk space of the system. The unit is KB. (Write-protected)</td>
</tr>
</tbody>
</table>

**COMMIT (This session only)**

When you click the button, the COMMIT command is executed only once. After you click the button, the following dialog box appears. Select drives to commit. When you click the **COMMIT and Reboot** button, the COMMIT command executed and the operating system reboots.

Click the **All Drives** button to enable COMMIT command on drive C through drive Z. (When all check boxes of drives are selected, click the **All Drives** button to disable COMMIT command on all the drives.)

When write-protect switch is enabled on models that has write-protected SSD, a warning message box appears and you cannot change the settings. You can only confirm the configuration.
Chapter 7  Terms of Use

7.1 Limited Warranty
Interface Corporation does not warrant uninterrupted or error-free operations of the software product. The entire risks as to the quality of or arising out of use or performance of the software products, if any, remains with you.

Interface believes that information contained in the document is accurate. The document is carefully reviewed for technical accuracy. Interface reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. Interface is not liable for any damages arising out of or related to this document or the information contained in it.

Charts and tables contained in this document are only for illustration purposes and may vary depending upon a user's specific application.

All official specifications are in metric. English unit is supplied for convenience.

7.2 Copyrights and Intellectual Property Rights
Interface Corporation owns all titles and intellectual property rights in and to the products. The products include the computer software, audio/visual content such as images, texts, or pictures.

7.3 Warning Regarding Medical and Clinical Use of Our Products
Our products are not designed for components intended to ensure a level of reliability suitable for use under conditions that might cause serious injury or death.

Our products are not designed with components and testing instrument intended to ensure a level of reliability suitable for use in treatment and diagnosis of human.

Applications of our products involving medical or clinical treatment can create a potential for accidental injury caused by product failure, or by errors on the part of the user or application engineer.

7.4 Prohibition of Reproduction
No part of this document may be reproduced or changed in any form without the prior consent of Interface Corporation.

7.5 Limitation of Liability
Interface Corporation will not be liable for any special, incidental, indirect or consequential damages whatsoever even if Interface Corporation or any reseller could foresee the possibility of damages.

Users shall assume any subsequent risks whatsoever resulting from such as using and installing this product.

Interface Corporation shall not be liable for any incidental or consequential damages, including damages or other costs resulting from defects which might be contained in the product, product supply delay or product failure.

Customer's right to recover damages caused by fault or negligence on the part of Interface Corporation shall be limited exclusively to product replacement.

This product is designed under Japanese domestic specifications. Interface Corporation is not responsible for the use of this product outside Japan. We do not offer any maintenance service or technical support abroad.

Interface Corporation is not liable for any damage arising from the included document or information.

7.6 Trademark
Products and company names are trademarks, registered trademarks, or servicemarks of their respective owners.

Copyright 2013 Interface Corporation. All rights reserved.