

File System Device I/O モジュール リファレンスガイド

第 1.01 版

2015 年 12 月

株式会社グレースシステム

[注意事項]

- すべての著作権は、株式会社グレープシステムにあります。
- 本ドキュメントの内容の一部または全部を無断で転載、複写、複製する事を禁じます。
- 本製品の仕様は予告なく変更される事があります。
- 本ドキュメントに記載されている会社名、製品名は各社の商標または登録商標です。

Copyright (C) 2007-2015 Grape Systems, Inc. All Rights Reserved

はじめに

本書は、File System と FSIF モジュールとの間に位置する Device I/O ドライバについて、GR-FILE 用に提供するファイルを用いて説明を記述したものです。

改訂履歴

Rev.	日付	改訂内容
1.00	2008 年 7 月	初版リリース
1.01	2015 年 12 月	Appendix A を追加

目次

1 File System Device I/Oモジュールの概要	1
1.1 File System Device I/Oモジュールとは	1
1.2 他モジュールとの関連	1
1.2.1 提供する機能	2
1.2.2 使用しているAPI	3
2 機能概要	4
grp_fsif_dev_io_init	5
grp_fsif_EventNotification	6
_grp_fs_grusb_open_dev	7
_grp_fs_grusb_close_dev	8
_grp_fs_grusb_read_dev	9
_grp_fs_grusb_write_dev	10
_grp_fs_grusb_ioctl_dev	11
3 他ファイルシステムの場合の注意事項	12
3.1 初期化处理	12
3.2 接続/切断通知処理	12
3.3 デバイスドライバインタフェース	12
Appendix A 内部構造	エラー! ブックマークが定義されていません。

API 関数索引

(アルファベット順)

_grp_fs_grusb_close_dev	8
_grp_fs_grusb_ioctl_dev	11
_grp_fs_grusb_open_dev	7
_grp_fs_grusb_read_dev	9
_grp_fs_grusb_write_dev	10
grp_fsif_dev_io_init	5
grp_fsif_EventNotification	6

1 File System Device I/Oモジュールの概要

File System Device I/O モジュールの概要について説明します。

なお本ドキュメントでは、GR-FILE 用に提供しています” grp_fsif_dev_io.c” を元に説明を記述いたします。

1.1 File System Device I/Oモジュールとは

File System Device I/O モジュールは、FSIF の提供する機能をご利用の File System のデバイスドライバに合わせる変換処理部です。

ご利用になる File System にあわせて本モジュールをご用意していただく必要がありますが、弊社が提供しています GR-FILE につきましては、” grp_fsif_dev_io.c” として提供しています。

1.2 他モジュールとの関連

以下に、GR-FILE 用の本モジュールの関連図を示します。

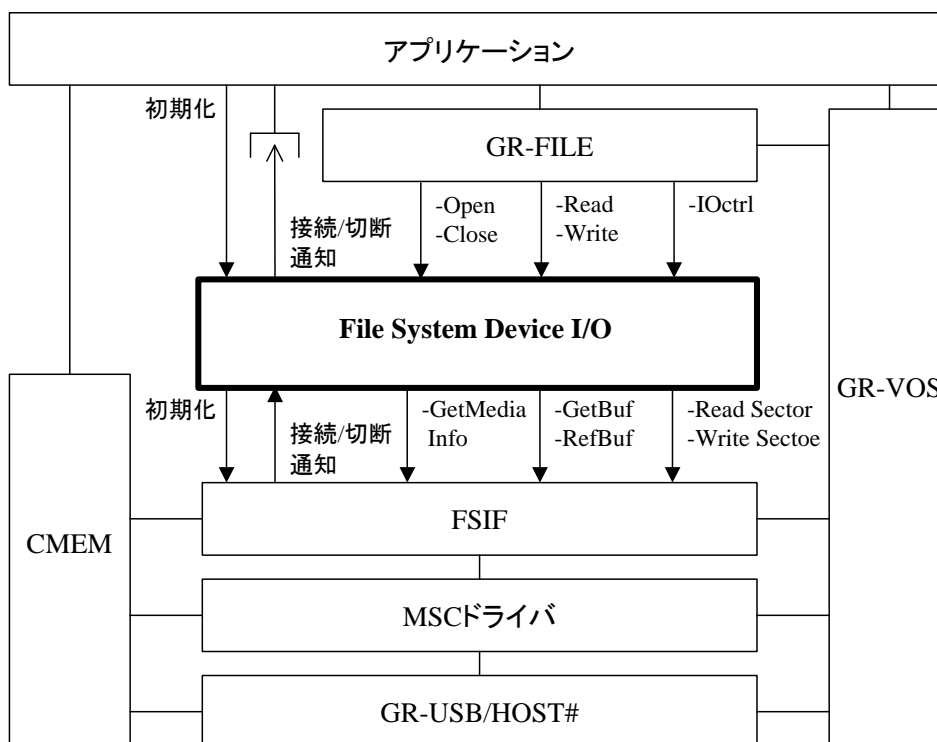


図 1-1 関連図

(1) GR-FILE

弊社の提供する FAT12/16/32 に対応したファイルシステムです。

(2) FSIF

弊社の提供する File System Interface モジュールです。セクタのリード/ライト処理、プラグアンドプレイ機能やマルチユニットデバイスのサポート、メディアチェックなどの機能を提供します。

(3) アプリケーション

File System Device I/O モジュールの初期化を行います。また、MS デバイスの接続/切断の通知を処理し、接続している間 **GR-FILE** を介して MS デバイスにアクセスすることが出来ます。

(4) MSC ドライバ

GR-USB/HOST#向け **MSC** (マストレージクラス) ドライバです。

(5) **GR-USB/HOST#**

USB に関する通信要求、ドライバ制御要求の API を提供するモジュールです。

(6) **CMEM**

CMEM (共通メモリ管理モジュール) は、**GR-USB/HOST#**で用いるスケジューリングリスト、制御データ・バッファ、アプリケーション・データ・バッファの領域を管理することを目的にした、共通インタフェースです。

(7) **GR-VOS**

仮想 OS モジュールです。

Grapeware は全て、**GR-VOS** の API を介して OS の資源を使用します。

ご使用される OS、環境向けにポーティングする必要があります。

1.2.1 提供する機能

” **grp_fsif_dev_io.c** ” にて提供している機能は以下のとおりです。

(1) 初期化処理

本モジュールおよび下位の **FSIF** モジュールを初期化します。

この時、接続/切断通知用のキューの生成および管理テーブルの初期化を行っていますが、この処理は **GR-FILE** 用の特別な処理となります。

(2) 接続/切断通知

メディアの接続/切断をキューにより通知します。アプリケーションでは、そのキューを受信する処理を実装していただく必要があります。キューについては「Appendix A OS資源」を参照してください。

通知はメディア毎に行いますので、複数の **I/F** を持つカードリーダーライタなどは複数回接続が通知されることがあります。また論理的にドライブを複数持つ場合も同様です。

(複数パーティションなどについては、本機能では関与しません)

なお、メディアの接続/切断をキューにより通知することは、**GR-FILE** の推奨する方法に基づいています。

(3) デバイスドライバインタフェース機能

GR-FILE にて規定されている、以下の 5 つのインタフェースを提供します。

- デバイスのオープン処理
- デバイスのクローズ処理
- デバイスのリード処理
- デバイスのライト処理
- デバイス固有の I/O 制御処理

提供しているインタフェースにつきましては、後述の機能概要の説明にて記述します。

1.2.2 使用しているAPI

以下に記す FSIF の API を利用しています。詳細については『GR-USB/HOST# FSIF モジュール API 仕様書』を参照ください。

grp_fsif_Init	FSIF の初期化
grp_fsif_WriteSector	データ書き込み処理
grp_fsif_ReadSector	データ読み込み処理
grp_fsif_GetMediaInfo	メディア情報取得処理
grp_fsif_GetNonCacheBuffer	CMEM 領域取得処理
grp_fsif_RelNonCacheBuffer	CMEM 領域解放処理

2 機能概要

本章では、” grp_fsif_dev_io.c” にて提供している以下の API について説明を記述します。定義や構造体などにつきましては、弊社 GR-FILE のドキュメントを参照してください。

初期化関数

grp_fsif_dev_io_init	本モジュールの初期化
----------------------	------------

コールバック通知

grp_fsif_EventNotification	接続/切断通知処理
----------------------------	-----------

デバイスドライバインタフェース

_grp_fs_grusb_open_dev	デバイスのオープン処理
_grp_fs_grusb_close_dev	デバイスのクローズ処理
_grp_fs_grusb_read_dev	デバイスの read 処理
_grp_fs_grusb_write_dev	デバイスの write 処理
_grp_fs_grusb_ioctl_dev	デバイス固有の I/O 制御処理

grp_fsif_dev_io_init

FileSystem Device I/O モジュールの初期化

【構文規則】

```
int grp_fsif_dev_io_init(void)
```

【入力パラメータ】

なし

【出力パラメータ】

なし

【返却値】

0	正常終了
-1	異常終了

【機能】

本関数では、ハンドル情報の管理(*1)の初期化および接続/切断通知用キューの生成を行います。
また、FSIF にてメディアの接続/切断を検知した際に呼び出される関数の登録をかねて FSIF の初期化を実行します。

【備考】

ハンドル情報の管理とは

FSIF ではメディアの接続を確認すると、上位に対してハンドル情報を提供します。切断の通知が発生するまで、このハンドル情報にてアクセスします。しかし GR-FILE ではデバイス番号と呼ばれる値を使用します。そのため、本モジュール内で配列として管理し、そのインデックス値を GR-FILE へ渡す処理を実装しています。

grp_fsif_EventNotification

接続/切断コールバック通知

【構文規則】

```
void grp_fsif_EventNotification( grp_s32 lEvtnt, void *pvHdr, grp_si iIdx)
```

【入力パラメータ】

lEvtnt	イベント
	GRP_FSIF_ATTACHED_MEDIA メディアの接続
	GRP_FSIF_DETACHED_MEDIA メディアの切断
pvHdr	ハンドル
iIdx	インデックス値（将来のため予約）

【出力パラメータ】

なし

【返却値】

なし

【機能】

FSIF にてメディアの接続を検知すると本関数がコールされます。ハンドル管理情報から空いている領域を検索します。設定値以上のメディアの接続の場合、エラーとして処理が終了します。空きがあった場合、「インデックス値+GRP_FSIF_ATTACHED_MEDIA (0x80000000)」をキューによりアプリケーションに通知します。

FSIF にてメディアの切断を検知すると本関数がコールされます（接続のコールバック通知があった場合のみ）。そして、ハンドル管理情報に一致するハンドル情報がない場合、処理を終了します。一致するハンドル情報があった場合、「インデックス値+GRP_FSIF_DETACHED_MEDIA (0)」をキューによりアプリケーションに通知します。

キューによる通知は、下位 1 バイトがインデックス値となります。

`_grp_fs_grusb_open_dev`

デバイスのオープン処理

【構文規則】

```
static int _grp_fs_grusb_open_dev( int iDev, int iRWOOpen, grp_int32_t *piHandle,  
                                   grp_uint32_t *puiOff, grp_uint32_t *puiSize, int *piSzShift)
```

【入力パラメータ】

iDev	デバイス番号
iRWOOpen	(未使用)

【出力パラメータ】

piHandle	(未使用)
puiOff	メディア内の開始オフセット
puiSize	指定デバイスのトータルブロック (セクタ) 数
piSzShift	指定デバイスの物理ブロック (セクタ) サイズの 2 べきシフト値

【返却値】

省略

【機能】

本関数では、`grp_fsif_GetMediaInfo` 関数にて接続しているメディアのセクタサイズ、およびトータルセクタ数を取得します。

次に接続されたメディアのパーティション情報の取得を行います。この時、**FSIF** で用意している `grp_fsif_GetNonCacheBuffer` 関数にてデータ領域を確保します。そしてリード処理にて先頭セクタの情報を取得します。取得した情報は `grp_fs_get_part()` にて解析し、パーティションの有無やパーティションタイプの確認の後に処理を終了します。

`_grp_fs_grusb_close_dev`

デバイスのクローズ処理

【構文規則】

```
static int _grp_fs_grusb_close_dev( grp_int32_t iHandle, int iDev)
```

【入力パラメータ】

iHandle	(未使用)
iDev	デバイス番号

【出力パラメータ】

なし

【返却値】

0	正常終了
---	------

【機能】

本関数では、特に実装すべき処理はありませんので、OK のみ返します。

`_grp_fs_grusb_read_dev`

デバイスのリード処理

【構文規則】

```
static grp_int32_t _grp_fs_grusb_read_dev( grp_int32_t iHandle, int iDev, grp_uint32_t  
                                         uiDevBlk, grp_uchar_t *pucBuf, grp_isize_t iCnt)
```

【入力パラメータ】

piHandle	(未使用)
iDev	デバイス番号
uiDevBlk	リード開始物理ブロックオフセット
iCnt	リードする物理ブロック数

【出力パラメータ】

pucBuf	リードしたデータを格納する領域のアドレス
--------	----------------------

【返却値】

正整数	実際に読み込んだ物理ブロック（セクタ）数
その他	エラー

【機能】

本関数では、`grp_fsif_ReadSector` 関数を使用しデータのリード処理を行います。デバイス番号からハンドル情報を取得し、他のパラメータはそのまま利用します。

`_grp_fs_grusb_write_dev`

デバイスのライト処理

【構文規則】

```
static grp_int32_t _grp_fs_grusb_write_dev( grp_int32_t iHandle, int iDev, grp_uint32_t  
                                           uiDevBlk, grp_uchar_t *pucBuf, grp_isize_t iCnt)
```

【入力パラメータ】

piHandle	(未使用)
iDev	デバイス番号
uiDevBlk	ライト開始物理ブロックオフセット
pucBuf	ライトするデータを格納した領域のアドレス
iCnt	ライトする物理ブロック数

【出力パラメータ】

なし

【返却値】

正整数	実際に書き込んだ物理ブロック（セクタ）数
GRP_FSIF_NG	エラー

【機能】

本関数では、`_grp_fs_grusb_write_de` 関数を使用しデータのライト処理を行います。デバイス番号からハンドル情報を取得し、他のパラメータはそのまま利用します

`_grp_fs_grusb_ioctl_dev`

デバイス固有の I/O 制御（未実装）

【構文規則】

```
static int _grp_fs_grusb_ioctl_dev( int iDev, grp_uint32_t uiCmd, void *pvParam)
```

【入力パラメータ】

iDev	デバイス番号
uiCmd	実行する I/O 制御の機能番号
pvParam	実行する I/O 制御に渡すパラメータ

【出力パラメータ】

pvParam	実行する I/O 制御に渡すパラメータ
---------	---------------------

【返却値】

GRP_FS_ERR_NOT_SUPPORT	未サポート
------------------------	-------

【機能】

本関数は未実装となります。

3 他ファイルシステムの場合の注意事項

本章では、GR-FILE 以外のファイルシステム用の File System Device I/O モジュールを作成するための注意事項について説明を記述します。

3.1 初期化処理

初期化では、以下の処理を行っています。

(1) File System Device I/O モジュール内部の初期化

- ・アプリケーションへの通知用キューの生成
- ・ハンドル情報の管理テーブルの初期化

(2) FSIF モジュールの初期化関数の実行

- ・接続/切断通知先コールバック関数の登録処理

(1) の処理は File System Device I/O モジュールをどのように実装し、またファイルシステムへの通知をどのように行うかに依存します。お使いの環境に合わせた処理を実装ください。

(2) の処理はメディアの接続の有無を確認するために必須の処理となります。必ず実装ください。

3.2 接続/切断通知処理

FSIF モジュールにてメディアの接続および切断検知しますと、上記(2)で登録した関数がコールされます (GR-FILE 用では `grp_fsif_EventNotification` 関数が該当します)。

パラメータにて接続および切断が判別できますので、必要な処理を実装してください。

3.3 デバイスドライバインタフェース

FSIF モジュールで提供している API を使い、ファイルシステムにあわせた実装をしてください。

もし FSIF モジュールで提供している API のみでは実装が難しい場合、別途、弊社サポートへお問い合わせください。

Appendix A OS資源

File System Device I/O モジュールで使用する OS 資源は以下のとおりです。

キュー : 1

(1) キュー

表 A-1キュー

#	名前	メッセージサイズ (バイト)	メッセージ数	用途
1	“qXFER”	4	10	接続/切断通知用

File System Device I/O モジュール リファレンスガイド

発行年月：2015 年 12 月 第 1.01 版

発行：株式会社グレープシステム

E-Mail : gr@support.grape.co.jp

URL : <http://www.grape.co.jp>

Copyright (C) 2007-2015 Grape Systems, Inc.

All rights reserved.