GR-USB/HOST#
FSIF モジュール
API 仕様書

第 1.01 版

2008 年 10 月

株式会社グレープシステム

[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。

はじめに

本書は、GR-USB/HOST# FSIF モジュールの API 仕様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	2008 年 10 月	全般 ・以下のコールバックに関するメンバ名変更による修正 grp_fsif_init_prm 構造体 pfnFsifNotification ← pfnEventNotification

目次

API 関数索引

（アルファベット順）

1 FSIF モジュールの概要

FSIF モジュールの概要について説明します。

1.1 FSIF とは

FSIF（File System InterFace）とは、上位ファイルシステムに通常のデバイスドライバとしての I/F を提供するモジュールです。
マルチユニットデバイスにも対応し、各メディア毎に情報を管理します。

1.2 他モジュールとの関連

以下に、FSIF と他モジュールとの関連および USB マスストレージ統合キット内の位置付けについて図示します。

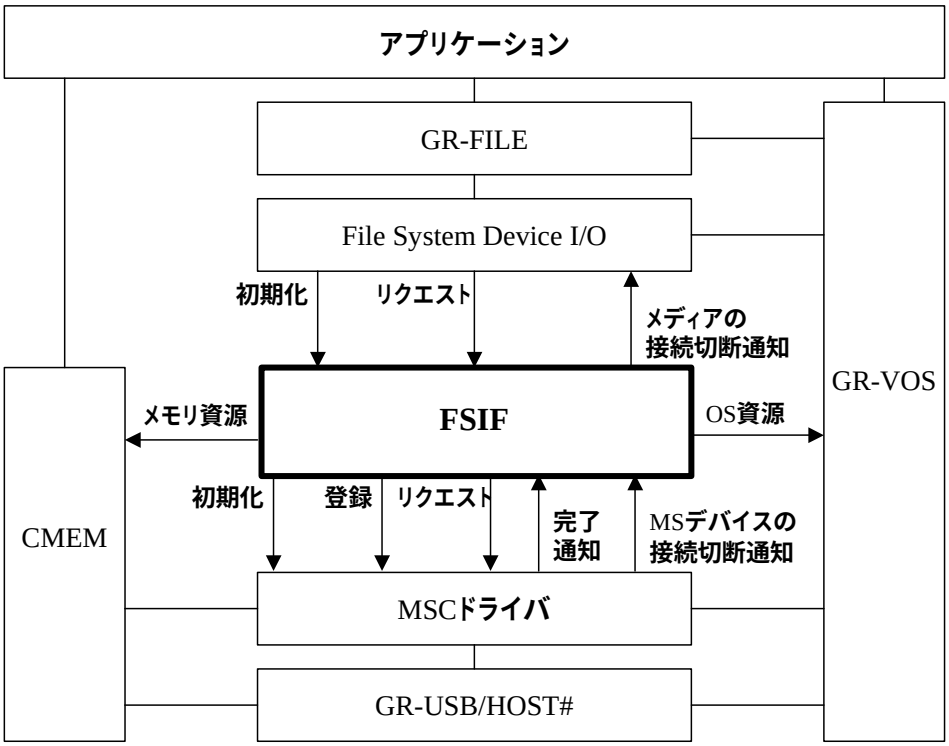

図 1-1 FSIF 関連図

（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）File System Device I/O

FSIF モジュールの I/F を GR-FILE 用の I/F に変換するラッパです。FSIF では“ grp_fsif_dev_io.c ”として提供しています。詳細につきましては『File System Device I/O モジュール リファレンスガイド』を参照ください。
※他のファイルシステムをご利用の場合、本ラッパ部を作成していただく必要があります。

（3）MSC ドライバ

GR-USB/HOST#向け MSC（マスストレージクラス）ドライバです。

（4）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。

（5）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、共通インターフェースです。

（6）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。

2 インターフェース

FSIF モジュールが提供するインターフェースについて説明します。

2.1 コーディング規約

Grapeware 製品では、以下の規約に準じてコーディングを行っております。

2.1.1 型名

環境に応じて変更できるよう、Grapeware 製品では一般的に使用する型を typedef 宣言を用いて別名に置き換えて使用しています。（grp_std_types.h）
Grapeware 製品で使用する型名は以下のとおりです。

表 2-1 Grapeware 製品で使用する型名

#	型名	内容
1	grp_ui	unsigned int
2	grp_si	signed int
3	grp_u32	unsigned long
4	grp_s32	signed long
5	grp_u16	unsigned short
6	grp_s16	signed short
7	grp_u8	unsigned char
8	grp_s8	signed char

2.1.2 変数名

変数や構造体のメンバなどは、変数の名称からその型がわかるようにするため、変数の先頭に次の識別用の文字を付けています。

表 2-2 変数名の先頭に付ける識別用文字

#	識別用文字	内容
1	g_	グローバル変数
2	l_	ローカル（Static）変数
3	p	ポインタ
4	a	配列
5	v	void
6	i	grp_si
7	ui	grp_ui
8	c	grp_s8
9	uc	grp_u8
10	s	grp_s16
11	us	grp_u16
12	l	grp_s32
13	ul	grp_u32
14	f	float
15	d	double

16	t	typedef（構造体など）
17	fn	関数
18	h	ハンドル

2.2 **エラーコード**

FSIF モジュールで使用するエラーコードについて説明します。

表 2-3 **エラーコード**

#	エラーコード	値	内容
1	GRP_FSIF_OK	0	正常終了
2	GRP_FSIF_NG	0x8600FFFF	異常終了
3	GRP_FSIF_VOS_ERROR	0x8600FFFE	GR-VOS 関連エラー
4	GRP_FSIF_CMEM_ERROR	0x8600FFFD	CMEM 関連エラー
5	GRP_FSIF_ILLEAGAL_ERROR	0x8600FFFC	不正エラー
6	GRP_FSIF_TMOUT	0x8600FFFB	タイムアウト
7	GRP_FSIF_PARAM_ERROR	0x8600FFFA	パラメータエラー
8	GRP_FSIF_QUEINFO_ERROR	0x8600FFF9	キュー情報取得・解放エラー
9	GRP_FSIF_ILLEAGAL_STATE	0x8600FFF8	不正状態
10	GRP_FSIF_BUSY	0x8600FFF7	ビジー
11	GRP_FSIF_CHECK_CONDITION	0x8600FFF6	RequestSense を必要とするエラー(内部使用)

2.3 **定義値**

FSIF モジュールで使用する定義値は以下のとおりです。

2.3.1 **イベントコード**

FSIF モジュールから上位へ通知する際に使用するイベントコードです。

表 2-4 **イベントコード**

#	定義値	値	内容
1	GRP_FSIF_ATTACHED_MEDIA	0x80000000	メディア接続
2	GRP_FSIF_DETACHED_MEDIA	0	メディア切断

2.3.2 **リセットモード**

FSIF モジュールにて提供するリセットの種別を表します。

表 2-5 **リセットモード**

#	定義値	値	内容
1	GRP_FSIF_MSC_RESET	1	マスストレージリセット
2	GRP_FSIF_RE_ENUMERATION	2	USB バスリセットおよびエニュメレーション

2.4 構造体

FSIF モジュールで使用する構造体は以下のとおりです。

2.4.1 grp_fsif_init_prm 構造体

表 2-6 grp_fsif_init_prm 構造体

#	メンバ名	型	内容
1	pfnFsifNotification	void (*)(grp_s32, void*, grp_si)	イベント通知用コールバック関数

2.4.2 grp_fsif_media_info 構造体

表 2-7 grp_fsif_media_info 構造体

#	メンバ名	型	内容
1	ulSectorSize	grp_u32	1 セクタサイズ（単位：バイト）
2	ulSectorNum	grp_u32	セクタ数
3	pucInquiry	grp_u8 *	INQUIRY データ
4	ucPeriDevType	grp_u8	FSIF 内部で使用
5	ucLun	grp_u8	メディアの論理ユニット番号
6	usUsbDevId	grp_u16	デバイス ID

2.5 API 関数一覧

FSIF モジュールは、以下の API 関数を用意しています。

初期化関数

grp_fsif_Init	FSIF モジュールの初期化

アクセス関数

grp_fsif_WriteSector	データ書き込み処理
grp_fsif_ReadSector	データ読み込み処理

メディア情報取得関数

grp_fsif_GetMediaInfo	メディア情報取得処理

リセット関数

grp_fsif_Reset	リセット処理

CMEM 領域管理関数

grp_fsif_GetNonCacheBuffer	CMEM 領域取得処理
grp_fsif_RelNonCacheBuffer	CMEM 領域解放処理

2.6 API 関数詳細

本項では API 関数の詳細なフォーマットについて記載しています。

grp_fsif_Init

FSIF モジュールの初期化

【構文規則】

grp_s32　　**grp_fsif_Init**(grp_fsif_init_prm* ptPrm)

【入力パラメータ】

ptPrm	初期化情報
― pfnFsifNotification	イベント発生時に呼び出されるコールバック関数 ※後述のコールバック関数参照

【出力パラメータ】

なし

【返却値】

GRP_FSIF_OK	正常終了
GRP_FSIF_NG	異常終了
GRP_FSIF_CMEM_ERROR	CMEM 関連エラー
GRP_FSIF_VOS_ERROR	GR-VOS 関連エラー

【機能】

本関数では FSIF モジュールの初期化を行います。
本関数を呼び出す前に、GR-VOS の初期化（grp_vos_Init）と GR-USB/HOST#の初期化（grp_usbc_HostInit）を実行する必要があります。
MSC ドライバの初期化（grp_msc_Init）や登録処理（grp_msc_Register）は FSIF モジュール初期化内で実行しますので、アプリケーションから直接実行する必要はございません。
また、GR-USB/HOST#の起動処理（grp_usbc_Enable）は、本関数実行後に呼び出してください。

メディアの接続／切断イベントが発生すると、設定したコールバック関数が呼ばれます。
なお、複数のインターフェースがあり、複数のマスストレージクラスがあるメディアに関しては、その数分のコールバック関数が呼ばれます。

【コールバック関数】

メディアの接続／切断イベントが発生した際に呼び出されるコールバック関数の形式は以下の通りです。

void (*pfnFsifNotification)(grp_s32 ulEvnt, void *vpHdr, grp_si iIdx)

＜出力パラメータ＞

ulEvnt	イベントコード
vpHdr	メディアハンドル
iIdx	（拡張用/未使用）

＜返却値＞

GRP_FSIF_OK を返してください。

イベントコードには、イベントの種類（接続／切断）がセットされます。イベントコードに関しましては「2.3.1 イベントコード」を参照ください。
メディアハンドルは、メディアの識別子です。切断されるまではデータの書き込みや読み込みなどで使用されますので、保存しておく必要があります。

grp_fsif_WriteSector

データ書き込み処理

【構文規則】

grp_s32 **grp_fsif_WriteSector**(void*　vpHdr, grp_u32　ulStartSector,
grp_u32　ulNumOfSector, grp_u8*　pucDataBuf)

【入力パラメータ】

vpHdr	メディアハンドル
ulStartSector	開始セクタ番号
ulNumOfSector	セクタ数
pucDataBuf	書込みデータが格納されているバッファの先頭アドレス（共通メモリ管理モジュール（CMEM）から確保した領域をご利用ください）

【出力パラメータ】

なし

【返却値】

GRP_FSIF_OK	正常終了
GRP_FSIF_NG	異常終了
GRP_FSIF_ILLEAGAL_ERROR	不正エラー
GRP_FSIF_TMOUT	タイムアウト
GRP_FSIF_PARAM_ERROR	パラメータエラー
GRP_FSIF_QUEINFO_ERROR	キュー情報取得・解放エラー

【機能】

本関数では、メディアハンドルによって指定されたメディアに対し、指定されたセクタ番号から、指定されたセクタ数分だけ、指定されたデータを書き込みます。
本関数からは処理が完了するまで戻りません。

grp_fsif_ReadSector

データの読み込み処理

【構文規則】

grp_s32 **grp_fsif_ReadSector**(void* vpHdr, grp_u32 ulStartSector,
grp_u32 ulNumOfSector, grp_u8* pucDataBuf)

【入力パラメータ】

vpHdr	メディアハンドル
ulStartSector	開始セクタ番号
ulNumOfSector	セクタ数
pucDataBuf	読み込みデータを格納するバッファの先頭アドレス（共通メモリ管理モジュール（CMEM）から確保した領域をご利用ください）

【出力パラメータ】

pucDataBuf	読み込んだデータ

【返却値】

GRP_FSIF_OK	正常終了
GRP_FSIF_NG	異常終了
GRP_FSIF_ILLEAGAL_ERROR	不正エラー
GRP_FSIF_TMOUT	タイムアウト
GRP_FSIF_PARAM_ERROR	パラメータエラー
GRP_FSIF_QUEINFO_ERROR	キュー情報取得・解放エラー

【機能】

本関数では、メディアハンドルによって指定されたメディアの指定されたセクタ番号から、指定されたセクタ数分だけデータを読み込み、指定されたバッファに格納します。
本関数からは処理が完了するまで戻りません。

grp_fsif_GetMediaInfo

メディア情報取得処理

【構文規則】

grp_s32　**grp_fsif_GetMediaInfo**(void*　vpHdr, grp_fsif_media_info*　ptMediaInfo)

【入力パラメータ】

vpHdr	メディアハンドル
ptMediaInfo	メディア情報を格納するエリアの先頭アドレス

【出力パラメータ】

ptMediaInfo	メディア情報
― ulSectorSize	1 セクタのサイズ（単位：バイト）
― ulSectorNum	総セクタ数
― pucInquiry	INQUIRY データが格納されているエリアの先頭アドレス
― ucLun	論理ユニット番号
― usUsbDevId	デバイス ID

【返却値】

GRP_FSIF_OK	正常終了
GRP_FSIF_ILLEAGAL_STATE	不正状態

【機能】

本関数では、メディアハンドルによって指定されたメディアに関する情報の取得を行います。
指定したメディアが接続状態にない場合は、GRP_FSIF_ILLEAGAL_STATE を返します。

grp_fsif_Reset

リセット処理

【構文規則】

grp_s32　**grp_fsif_Reset**(void *vpHdr, grp_u32 ulMode)

【入力パラメータ】

vpHdr	メディアハンドル
ulMode	リセットモードの指定 GRP_FSIF_MSC_RESET マスストレージリセット GRP_FSIF_RE_ENUMERATION USB バスリセットとエニュメレーション

【出力パラメータ】

なし

【返却値】

GRP_FSIF_OK	正常終了
GRP_FSIF_ILLEAGAL_STATE	不正状態

【機能】

本関数では、リセットモードにより指定されたリセットを実行します。

grp_fsif_GetNonCacheBuffer

CMEM 領域の取得

【構文規則】

grp_s32　**grp_fsif_GetNonCacheBuffer**(grp_u8 **ppucBuf)

【入力パラメータ】

ppucBuf	取得したバッファ領域の先頭アドレスを格納するエリア

【出力パラメータ】

ppucBuf	取得したバッファ領域の先頭アドレス

【返却値】

GRP_FSIF_OK	正常終了
GRP_FSIF_NG	異常終了
GRP_FSIF_CMEM_ERROR	CMEM 関連エラー

【機能】

本関数では、CMEM（共通メモリ管理モジュール）を用いて非キャッシュ領域からバッファ領域を取得します。また、バッファ領域の排他制御も行います。

【備考】

USB マスストレージ統合キットでは、ブートセクタを読み込む際のバッファ領域を、本関数で確保しています。従って、2Kbyte のメモリブロックを 1 つしか用意しておりません。
もし、本関数をブートセクタの読み込み以外で使用される場合は、CMEM の該当するメモリパーティションプール（GRP_CMEM_ID_FSIF_NC_BUF）のメモリブロックサイズとメモリブロック数を、環境に適した値に設定し直して頂く必要があります。

grp_fsif_RelNonCacheBuffer

CMEM 領域の解放

【構文規則】

grp_s32 **grp_fsif_RelNonCacheBuffer**(grp_u8 *pucBuf)

【入力パラメータ】

pucBuf	解放するバッファ領域の先頭アドレス

【出力パラメータ】

なし

【返却値】

GRP_FSIF_OK	正常終了
GRP_FSIF_NG	異常終了
GRP_FSIF_CMEM_ERROR	CMEM 関連エラー

【機能】

本関数では、CMEM（共通メモリ管理モジュール）を用いて非キャッシュ領域にバッファ領域を解放します。また、バッファ領域の排他制御も行います。

GR-USB/HOST#
FSIF モジュール
API 仕様書

発行年月：2008 年 10 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