

組込み向けファイルシステム
GR-FILE
(概要説明書)

第1.22版

2011 年 6 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) 2003-2011 Grape Systems, Inc. All Rights Reserved

1. 概要

GR-FILE は、組込みシステムでファイルアクセスを実現するためのミドルウェアです。本章では、まず、**GR-FILE** の位置づけ、及び、機能・特徴概要について説明します。

1.1 GR-FILEの位置づけ

GR-FILE は、ディスクやメモ리카ード上に構築されたファイルシステム内のファイルをアクセスするためのミドルウェアです。**GR-FILE** は、図 1-1 に示しますように、ディスクやメモ리카ードを物理ブロック単位でI/Oするデバイスドライバと、アプリケーションプログラムの間に位置します。

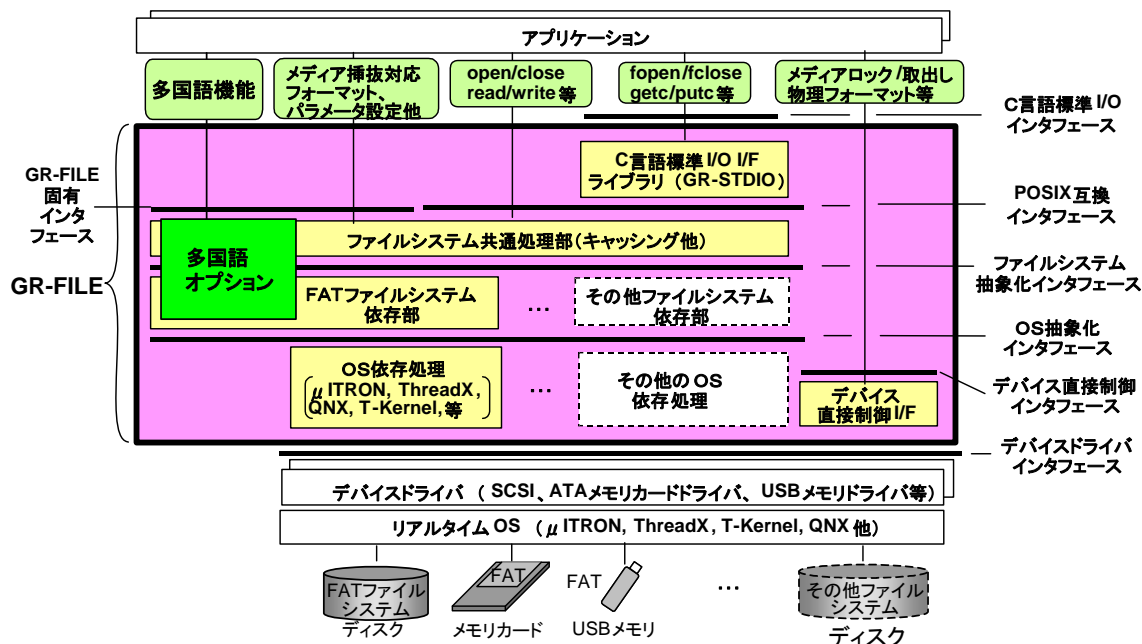


図 1-1 GR-FILE の位置づけ

GR-FILE は、ファイルシステム依存部を分離し、様々なファイルシステムをサポートできるようにになっています。本バージョンの **GR-FILE** は、ファイルシステムとして、PC 等で利用されている FAT ファイルシステムをサポートしています。また、OS 依存部分を分離し、様々なリアルタイム OS に簡単に対応可能な形になっています。μITRON 準拠の OS については、同 OS に対応した依存コードを提供可能です。さらに、OS レス環境にも適用可能です。

アプリケーションインタフェースとしては、POSIX 準拠の open/close/read/write 等を使ったインタフェースを提供しますと共に、fopen/fclose/getc/putc 等を使った C 言語の標準 I/O インタフェースの利用が可能です。また、メディアの挿抜処理対応、フォーマット、キャッシングサイズ等の各種パラメータ設定等の **GR-FILE** 固有のインタフェース、メディアの取出し/ロック/アンロック、物理フォーマット機能等のデバイス固有で提供された機能を利用するためのデバイス直接制御インタフェースも提供します。

GR-FILE は、100% ANSI C 言語で記述されており、組込みアプリケーションとリンクして利用頂くことができます。

表 1-1 に **GR-FILE** とその周りの各コンポーネントの概要を示します。また、表 1-2 に **GR-FILE** が提供または利用するインタフェースの概要を示します。

表 1-1 GR-FILE 及びその周りのコンポーネントの概要

#	コンポーネント	機能概要・要件
1	アプリケーション	<ul style="list-style-type: none"> ・組込みシステムの各種アプリケーションプログラム ・アプリケーションプログラムは GR-FILE が提供する C 言語標準 I/O インタフェース、POSIX 互換インタフェース、GR-FILE 固有インタフェース、デバイス直接制御インタフェースを使い、ファイル I/O を実現 ・特別なシステムアプリケーションにおいて、メディアの挿抜処理を実現。本挿抜処理では、GR-FILE が提供するインタフェースを使用し、デバイスドライバからのメディアの挿抜通知等を受け、ファイルシステムの自動 mount/unmount 処理等を実行
2	C 言語標準 I/O インタフェースライブラリ	<ul style="list-style-type: none"> ・fopen/fclose/getc/putc 等の C 言語の標準 I/O インタフェースをアプリケーションに対して提供 ・本ライブラリは、GR-FILE のオプションとして提供
3	ファイルシステム共通処理部	<ul style="list-style-type: none"> ・ファイルシステム非依存のアプリケーションインタフェースを提供 <ul style="list-style-type: none"> ①POSIX 互換インタフェース(open/close/read/write 等) ②GR-FILE 固有インタフェース(メディア挿抜対応処理、パラメータ設定等) ・ファイルデータやファイル管理情報のキャッシング処理機能を提供 ・GR-FILE で規定したファイルシステム抽象化インタフェースを用い、ファイルシステム依存処理を実行（ファイルシステムテーブルに設定されたインタフェース関数を実行）
4	ファイルシステム依存処理部 (FAT ファイルシステム依存部等)	<ul style="list-style-type: none"> ・ファイルシステムに依存した処理を実行 ・各ファイルシステム依存部は、GR-FILE で規定したファイルシステム抽象化インタフェースを提供し、同関数をファイルシステムテーブルに登録することで、ファイルシステム共通処理部からコールされる ・本バージョンでは、FAT ファイルシステムをサポート ・その他のファイルシステムを利用する場合は、GR-FILE で規定したファイルシステム抽象化インタフェースを提供する同ファイルシステム依存処理部の作成が必要
5	OS 依存処理部	<ul style="list-style-type: none"> ・セマフォ、メモリ割当て/解放、現在時刻取得、タスク ID の取得、タスク固有空間とのデータ授受等の OS/プラットフォーム依存の処理を実行 ・本バージョンでは、μITRON の OS 依存部をサポート ・その他の OS (含む OS レス) では、GR-FILE で規定した OS 抽象化インタフェースを提供する OS 依存処理部の作成が必要 ・OS レス環境等のために簡易メモリ管理ライブラリを GR-FILE で提供
6	デバイスドライバ	<ul style="list-style-type: none"> ・GR-FILE で規定したデバイスドライバインタフェース(open/close/物理ブロック単位の read/write)に従い、ディスクやメモリカード等のメディアへの物理ブロック単位での I/O 機能を GR-FILE に対して提供 ・メディアの挿抜処理を行うシステムアプリケーションに対し、メディアの挿抜等のイベント通知を実行 ・オプションで、メディアの取出し/ロック、物理フォーマット等、デバイス固有の制御機能を提供 ・本バージョンでは、RAM 上の仮想ディスクを使った GR-FILE とのインタフェース関数のサンプルを提供
7	リアルタイム OS	<ul style="list-style-type: none"> ・セマフォ、メモリ割当て/解放機能、現在時刻取得、タスク ID 取得等の機能を GR-FILE に提供 ・可変長のメモリ割当て/解放機能がない場合は、GR-FILE が提供する簡易メモリ管理ライブラリを利用可能 ・OS レスまたはシングルタスク環境では、セマフォ、タスク ID 取得機能はダミー処理で可
8	多国語オプション	<ul style="list-style-type: none"> ・GR-FILE に、多国語ファイル名へのアクセス、および、多数ファイルへの高速アクセスを実現するための GR-FILE 用オプション

注) # 2、# 3、# 4、# 5 が **GR-FILE** で提供するコンポーネントです。# 1、# 6、# 7 は、**GR-FILE** とインタフェースを持つコンポーネントに対して、**GR-FILE** が前提する機能の概要を示しています。

8 は、**GR-FILE** に多国語機能を付加するオプションです。

表 1-2 GR-FILE のインタフェース概要

#	インタフェース	形態	概要
1	POSIX 互換 インタフェース	提供	<ul style="list-style-type: none"> ・ GR-FILE からアプリケーションに対するインタフェース ・ open/close/read/write 等 LINUX/Windows で提供されている POSIX 互換 I/O インタフェース相当 ・ 実インタフェースは、"grp_fs_XXX", "GRP_FS_XXX"等の名称を持ち、"grp_include/grp_fs_conv.h"をインクルードすることで、POSIX 互換のインタフェース名でアクセス可能
2	C 言語標準 I/O インタフェース	提供	<ul style="list-style-type: none"> ・ GR-FILE からアプリケーションに対するインタフェース ・ GR-FILE のオプションライブラリ GR-STDIO により提供 ・ C 言語標準 I/O の fopen/fclose/getc/putc 等のインタフェースを提供 ・ 実インタフェースは、"grp_stdio_XXX" 等の名称を持ち、<stdio.h>の代わりに"grp_include/grp_stdio.h"をインクルードすることで、C 言語標準 I/O 互換のインタフェース名でアクセス可能
3	GR-FILE 固有 インタフェース	提供	<ul style="list-style-type: none"> ・ GR-FILE からアプリケーションに対するインタフェース ・ GR-FILE のメディアの挿抜対応処理インタフェースや、パラメータ設定等のインタフェースを提供
4	デバイス直接制御 インタフェース	提供	<ul style="list-style-type: none"> ・ GR-FILE からアプリケーションに対するインタフェース ・ メディアの取出し/ロック、物理フォーマット等、デバイス固有で提供された機能等をアプリケーションから利用するインタフェースを提供
5	ファイルシステム 抽象化インタフェース	内部 I/F	<ul style="list-style-type: none"> ・ GR-FILE 内のファイルシステム共通処理部とファイルシステム依存部間の内部インタフェース ・ 同インタフェースに従い、新たなファイルシステムに対応したファイルシステム依存処理関数をファイルシステムテーブルに登録することで、FAT 以外のファイルシステムもアクセスが可能となる
6	OS 抽象化 インタフェース	内部 I/F	<ul style="list-style-type: none"> ・ GR-FILE と OS/プラットフォーム依存部分のインタフェース ・ セマフォ、メモリ割当て/解放、現在時刻取得、タスク ID の取得、タスク固有空間とのデータの授受等 ・ μITRON 準拠の OS については、同インタフェースに対応した OS 依存部を GR-FILE でサポート ・ その他の OS については、同インタフェースに準拠した OS 依存部の作成が必要
7	デバイスドライバ インタフェース	利用	<ul style="list-style-type: none"> ・ GR-FILE がデバイスドライバに対して規定したインタフェース ・ デバイスの open/close/物理ブロック単位の read/write のインタフェースを規定 ・ 本インタフェースに従ったデバイスドライバの作成が必要 ・ 本バージョンでは、μITRON のデバイスドライバを使った GR-FILE とのインタフェース関数のサンプルを提供 ・ なお、メディアの挿抜に伴うファイルシステムの自動 mount/unmount 処理は、本デバイスドライバインタフェースでは規定しない。同処理については、デバイスドライバでメディアの挿抜を検出して、同イベントを挿抜処理用のシステムアプリケーションに伝え、同システムアプリケーションが、GR-FILE が提供する#3 のインタフェースを使用して実現する ・ GR-FILE では、挿抜処理用のシステムアプリケーションのサンプルを提供 ・ #6 のデバイス直接制御インタフェース等で利用する、デバイス固有の制御機能をオプションでサポート

1.2 機能・特徴概要

表 1-3 に **GR-FILE** の機能・特徴を示します。(ピンクの網掛け項目が特に **GR-FILE** の他にない特徴)

表 1-3 GR-FILE の機能・特徴一覧

#	項目	機能・特徴	効果
1	FAT ファイルシステムサポート	・ FAT12/16/32, ロングファイル名をサポート	・ PC とのデータ交換が可能
2	標準+固有アプリケーションインタフェース	<ul style="list-style-type: none"> ・ POSIX 準拠 I/O インタフェース open/close/read/write 等 ・ C 言語標準 I/O インタフェース(オプション) fopen/fclose/getc/putc 等 ・ GR-FILE 固有インタフェース メディア抜き取り対応処理、フォーマット/パーティション設定、パラメータ設定等 ・ デバイス直接制御インタフェース メディアの取出し/ロック/アンロック、物理フォーマット等、デバイス固有機能の利用インタフェース 	<ul style="list-style-type: none"> ・ LINUX/Windows 用ソースの利用、同 OS 上でのテストが可能 ・ アプリケーション開発が容易 ・ メディア、プラットフォームに応じた最適処理が可能 ・ デバイス固有機能利用が可能
3	マルチタスク I/O	・ 複数タスクによる同時 I/O をサポート	・ マルチタスク環境で利用可能
4	日本語(多国語)ファイル名対応	<ul style="list-style-type: none"> ・ Shift-JIS ファイル名をサポート ・ 他の日本語コード、多国語の対応も容易 ・ Shift-JIS 用コードの取外しも可能 	<ul style="list-style-type: none"> ・ 日本語システムに即適用可能 ・ 他国語システムにも対応可能 ・ コードサイズの削減が可能
5	ドライブ型+階層化 mount	・ ドライブ毎に個別のファイルシステムとする方法に加え、ファイルパス上に別ファイルシステムを接続する階層型 mount をサポート	・ 複数デバイスを 1 つの論理ファイルシステムに見せ、各デバイスサイズの制限を排除可能
6	データと管理情報のキャッシング(高速化)	・ ファイルデータとファイル管理ブロックの情報を分離してキャッシュし、さらに、オープン中のファイル情報、フリーブロック、ファイル名情報も別途キャッシング	・ 大容量ファイルの I/O に対しても、ファイル領域割当て情報をキャッシュ上に維持でき、高速 I/O が可能
7	キャッシュの write 制御(信頼性)	<ul style="list-style-type: none"> ・ メディアの特性に応じ、キャッシュ上の最新情報のメディアへの反映方式を選択可能 (1) write through 方式 各 write 時に変更をメディアに反映 (2) each close 方式 ファイル close 時にメディアに反映 (3) last close 方式 当該ファイルシステムの最後のファイルに対する close 時に反映 (4) unmount 方式 メディア unmount 時にメディアに反映 	<ul style="list-style-type: none"> ・ メディアの特性に応じて、性能と信頼性をバランスよく実現可能 (例 1) 内蔵ハードディスク 取外しができないため、性能を重視し、unmount 方式を採用 (例 2) メモリカード 不意の取外しがあるため、信頼性を考え、各アプリケーション終了時に確実にデータの反映を行う each close 方式を採用
8	アプリケーションバッファとメディア間の直接/連続ブロック I/O	・ キャッシュにないデータに対し、キャッシュバッファを使用せず、アプリケーションバッファとメディア間で直接、かつ、連続したブロックを一括して I/O するモードを選択可能	・ 大容量データで、頻繁には read しないファイルと、少量データで頻繁に read するファイルを区別して高速な I/O を実現可能。
9	ファイルシステム依存部の分離	・ キャッシング等のファイルシステムに共通な処理と、ファイルシステム依存処理部を分離	・ 同一インタフェースで、様々なファイルシステムをサポート可能
10	OS 依存処理部の分離	<ul style="list-style-type: none"> ・ OS/プラットフォーム依存処理部分をファイルシステム処理から分離。 ・ μITRON 準拠 OS 用の依存処理をサポート 	<ul style="list-style-type: none"> ・ 様々な OS への移植が容易 ・ OS レス環境下での利用も可能
11	メディアの挿抜対応	<ul style="list-style-type: none"> ・ キャッシュ write 制御により反映契機を制御 ・ 不意の取外し以降の I/O 抑止機能を提供 ・ メディア再挿入時のボリューム名チェック機能を提供 ・ メディアへの反映ができなくなったキャッシュデータの読出し機能を提供 	・ メモリカード等で、不意の挿抜が予想されるメディアに対し、メディア上のデータの整合性確保、間違ったメディアへの書込みの防止が可能

12	メディアの フォーマット/ パーティション 設定	<ul style="list-style-type: none">・メディアサイズに応じて FAT12/16/32 を自動的に選択しメディアのフォーマットが可能・FAT タイプ、クラスタサイズの明示指定も可・パーティションの設定変更も可能	<ul style="list-style-type: none">・自動簡易フォーマットとカスタマイズフォーマットの両方が可能・パーティションのフォーマットに加え、パーティション設定が可能
13	RAM ディスク 機能の提供	<ul style="list-style-type: none">・特別なデバイスメディアの1つとしてメモリ上の仮想ディスク機能を提供	<ul style="list-style-type: none">・メモリ上に高速なファイルシステムを構築可能
14	各種パラメータの 設定・変更	<ul style="list-style-type: none">・キャッシュブロックサイズ、キャッシュブロック数、同時オープンファイル数等の各種パラメータを実行時の初期化处理で設定・変更が可能	<ul style="list-style-type: none">・ターゲットシステムの構成に応じ、同ターゲットシステムに適した各種パラメータを実行時に設定可能

2. アプリケーションインタフェース

GR-FILEで提供します3種類のアプリケーションインタフェースの一覧を、表2-1、表2-2、表2-3、に示します。

表2-1 POSIX 互換アプリケーションインタフェース一覧

#	関数名	機能	関連実関数名
1	chdir	・カレントディレクトリの設定、及び、無効化	grp_fs_chdir
2	chmod	・ファイル保護モードの設定・変更	grp_fs_chmod
3	close	・ファイルのクローズ	grp_fs_close
4	closedir	・opendir でオープンしたディレクトリのクローズ	grp_fs_closedir
5	creat	・ファイルの作成	grp_fs_open
6	ftruncate	・指定サイズ以降のファイル領域の解放	grp_fs_ftruncate
7	getcwd	・カレントディレクトリ名の取得	grp_fs_get_cwd
8	lseek	・ファイルの読書き位置の設定・変更	grp_fs_lseek
9	mkdir	・ディレクトリの作成	grp_fs_create
10	open	・ファイルのオープン	grp_fs_open
11	opendir	・readdir でディレクトリエントリ読み出しを行うためのディレクトリのオープン	grp_fs_opendir
12	read	・ファイルの読込み	grp_fs_read
13	readdir	・ディレクトリエントリの読み出し	grp_fs_readdir
14	rename	・ファイル/ディレクトリ名称の変更	grp_fs_rename
15	rmdir	・ディレクトリの削除	grp_fs_unlink
16	stat	・ファイル属性情報の取得	grp_fs_stat
17	sync	・未反映キャッシュデータのメディアへの書戻し	grp_fs_sync
18	unlink	・ファイルの削除	grp_fs_unlink
19	utimes	・ファイル/ディレクトリのアクセス・更新時刻情報の変更	grp_fs_utimes
20	write	・ファイルの書込み	grp_fs_write

表2-2 C 言語標準 I/O インタフェース一覧

#	関数名	機能
1	clearerr	・エラー状態の解除
2	fclose	・ファイルのクローズ
3	feof	・ファイルの終わり判定
4	ferror	・エラー情報の取得
5	fflush	・I/O バッファ上のデータの書戻し/先読みデータのクリア
6	fgetc/ getc	・ファイルからの一文字 read
7	fgets	・ファイルからの一行 read
8	fileno	・ファイルハンドルの取得
9	fopen	・ファイルの open
10	fprintf/vfprintf	・ファイルへの書式付 write
11	fputc/putc	・ファイルへの一文字 write
12	fread	・ファイルからの read
13	fseek	・ファイルの読書き位置変更
14	ftell	・ファイルの読書き位置情報取得
15	fwrite	・ファイルへの write
16	getchar	・標準入力からの一文字 read
17	putchar	・標準出力への一文字 write
18	rewind	・ファイル読書き位置のファイル先頭への移動
19	ungetc	・入力の一文字ロールバック

表 2-3 GR-FILE 固有 I/O インタフェース一覧

#	関数名	機能	備考	最小構成*2
1	grp_fat_find_type	・メディア/パーティションのフォーマットパラメータ計算	フォーマット関連	○
2	grp_fat_format	・メディア/パーティションの初期化	フォーマット関連	○
3	grp_fat_format_sd *1	・SD カードのフォーマット (パーティション設定を含む)	参考ライブラリ	×
4	grp_fs_chdir	・カレントディレクトリの設定、及び、無効化	POSIX 対応	○
5	grp_fs_check_fs_dev	・再挿入メディアのチェック	挿抜処理対応	×
6	grp_fs_check_volume	・未 mount メディアのボリューム名の取得	挿抜処理対応	×
7	grp_fs_chmod	・ファイル保護モードの設定・変更	POSIX 対応	×
8	grp_fs_close	・ファイルのクローズ	POSIX 対応	○
9	grp_fs_closedir	・grp_fs_opendir でオープンしたディレクトリのクローズ	POSIX 対応	×
10	grp_fs_creat	・ファイル/ディレクトリの作成	mkdir 相当	○
11	grp_fs_err	・エラー番号/メッセージ変換	ライブラリ	×
12	grp_fs_get_attr	・ファイル属性情報の取得	stat 相当	○
13	grp_fs_get_cwd	・カレントディレクトリ名の取得	POSIX 対応 ライブラリ	×
14	grp_fs_get_dirent	・ディレクトリエントリ情報の取得	readdir, _findnext 相当	○
15	grp_fs_get_error	・未反映キャッシュバッファの読出し・解放	挿抜処理対応	×
16	grp_fs_get_mnt	・全 mount 情報の取得	getmntinfo 相当	×
17	grp_fs_get_mnt by_dev	・デバイス番号による特定 mount 情報の取得	getmntinfo 相当	×
18	grp_fs_get_mnt by_name	・デバイス名称による特定 mount 情報の取得	getmntinfo 相当	○
19	grp_fs_init	・GR-FILE の初期化	初期化处理	○
20	grp_fs_invalidate _fs_dev	・メディアに対する I/O 抑止設定 (メディア不当取出し時)	挿抜処理対応	×
21	grp_fs_lookup_dev	・デバイス名称→デバイス番号変換	その他関数	○
22	grp_fs_lseek	・ファイルの読書き位置の設定・変更	POSIX 対応	○
23	grp_fs_mount	・メディアの mount 処理 (メディア挿入時)	挿抜処理対応	○
24	grp_fs_open	・ファイルのオープン	POSIX 対応	○
25	grp_fs_opendir	・grp_fs_readdir でディレクトリエントリ読み出しを行うためのディレクトリのオープン	POSIX 対応	×
26	grp_fs_read	・ファイルの読込み	POSIX 対応	○
27	grp_fs_readdir	・ディレクトリエントリの読み出し (POSIX 互換用)	POSIX 対応	×
28	grp_fs_read_part	・メディアからのパーティション情報の読み込み	フォーマット関連	×
29	grp_fs_rename	・ファイル/ディレクトリ名称の変更	POSIX 対応	○
30	grp_fs_set_attr	・ファイル属性情報の設定・変更	chmod, utimes 対応	○
31	grp_fs_stat	・ファイル属性情報の取得	POSIX 対応	×
32	grp_fs_sync	・未反映キャッシュデータのメディアへの書戻し	POSIX 対応	○
33	grp_fs_task_ free_all_env	・全タスクのオープンファイル・カレントディレクトリの無効化	タスク管理	×
34	grp_fs_task_ free_env	・自タスクのオープンファイル・カレントディレクトリの無効化	タスク管理	×
35	grp_fs_task_ free_env by_id	・指定タスクのオープンファイル・カレントディレクトリの無効化	タスク管理	×
36	grp_fs_truncate	・指定サイズ以降のファイル領域の解放	POSIX 対応	○
37	grp_fs_unlink	・ファイル/ディレクトリの削除	POSIX 対応	○

38	grp_fs_unmount	・メディアの unmount 処理	挿抜処理対応	○
39	grp_fs_utimes	・ファイル/ディレクトリのアクセス・更新時刻情報の変更	POSIX 対応	×
40	grp_fs_write	・ファイルの書き込み	POSIX 対応	○
41	grp_fs_write_part	・メディアへのパーティション情報の書き込み	フォーマット関連	×

*1 SD Card Association の規格を含むため、製品に利用する場合は、SD Card のライセンスが必要となります。

*2 コンパイルスイッチ「GRP_FS_MINIMIZE_LEVEL」で最小構成を行った場合の関数の使用制限を表します。

組込み向けファイルシステム GR-FILE

発行年月：2011 年 6 月 第1.22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) 2003-2011, Grape Systems, Inc.

All rights reserved.