

## 第4章

# 組み込みシステム向け FAT ファイル・システム FatFs と極小版 Petit-FatFs の使い方

赤松 武史

ごく小規模な組み込みシステムにおいても、ファイル・システムは重要な要素部品になってきた。ここでは、筆者が開発した組み込み向け汎用 FAT ファイル・システム FatFs の使い方について詳しく解説する。また、RAM 50 バイト程度でも動作する Petit-FatFs についても取り上げる。  
(筆者)

## 1 FatFs の概要

汎用 OS はファイル・システムの上に構築されているのに対し、OS なしまたはリアルタイム OS (RTOS) ベースの小規模組み込みシステムの多くはファイル・アクセス機能が必要としませんでした。多くの RTOS にはファイル・システムが存在しないか、あってもオプション扱いです。このため、メモリ・カードなどのメディアにアクセスする機能が必要になった場合は、適当なファイル・システム(ミドルウェア)を入手するか作成して実装しなければなりません。

FatFs は、そのようなシステムをターゲットとして開発した汎用 FAT ファイル・システムです(図 1)。当初 8 ビット・マイコンでファイル・アクセス機能を実現するための

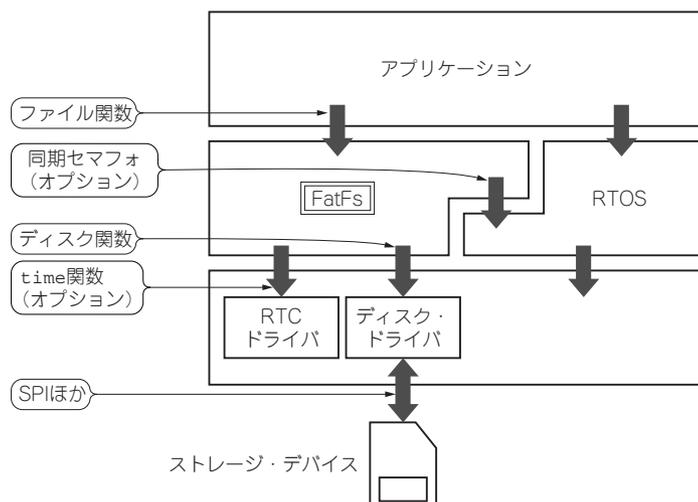


図 1 組み込みシステムとファイル・システム

コンパクトなミドルウェアを目指してプロジェクトを開始しました。しかし、実際には 32 ビット・マイコンで RTOS とともに使われることも多く、求められる機能も増えてきたため、OS 指向な機能拡張を続けながら現在に至っています。なお、ここで言うファイル・システムとは、ファイル API を提供するミドルウェアを指すものとしします。

### ● FatFs の特徴と機能

FatFs には次に示す特徴があり、多くの機能はコンパイル時の構成オプションにより機能変更・選択が可能です。

- Windows 互換 FAT ファイル・システム
- プラットホームに依存しない
- コンパクトなコード・サイズと RAM 使用量
- 複数のボリュームに対応(物理ドライブ・区画)
- 複数の ANSI/OEM コード・ページに対応(ファイル名)
- 長いファイル名に対応

(Unicode API も選択可)

- RTOS に対応  
(再入、ファイル共有制御)

FatFs の提供する機能を表 1 の API リストに示します。また、本稿では FatFs のリビジョンは R0.08 を想定して解説していきます。

### ● FatFs の移植に必要なこと

FatFs 自体は C89 に準拠して書かれていて、プラットフォーム依存部が完全に分離されています。必要な機能に従って構成オプションを設定し、コンパイルが問題なく通ればその処理系での使用は可能です。コード・サイズは、標準構成(リード/ライト/シークおよび各種ディレクトリ操作機能)において、8 ビット CPU で 11K ~ 16K バイト、