オペレーティングシステム
第１５章 FAT ファイルシステム

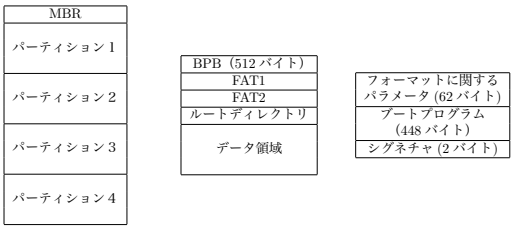
https://github.com/tctsigemura/OSTextBook

FAT ファイルシステム

- 1980 年代から PC のファイルシステムとして利用されてきた．
- 仕様が公開されているので様々なところで利用されている．
  - USB メモリ，SD カード，その他メモリカード
  - Windows，Linux，macOS は FAT ファイルシステムをサポート
    → 同じ USB メモリを使用してデータ交換が可能
  - デジカメ，音楽プレーヤ，デジタルテレビ，カーナビ...
    → デジカメの SD カードを PC で読める．
- ファイル名は半角 8 文字＋ 3 文字（例：`IMG_1234.JPG`）
- FAT にはいくつかの種類がある．

種類	最大ボリュームサイズ	最大ファイルサイズ	ファイル名
FAT12	32MiB	32MiB	8+3 文字
FAT16	2GiB	2GiB	8+3 文字
FAT32	2TiB	4GiB	8+3 文字
exFAT	16EiB	16EiB	255 文字

ボリューム内部の構造



- （ディスク装置全体，または，パーティション）＝ ボリューム
- パーティションの位置と大きさは MBR のテーブルで決まった．
- ボリュームの内部構造は FAT ファイルシステムのルールで決まる．
- ボリューム内部のコンポーネントの大きさなどは BPB から分かる．
- FAT は大切なデータなので２重化してある．
- データ領域は**クラスタ**と呼ばれるブロック単位で扱う．

BPB（BIOS Parameter Block）

パラメータ	意味	位置	長さ	値の例
ジャンプ命令	ジャンプ機械語命令	0	3	`0xeb 0x3e 0x90`
セクタサイズ	1 セクタのバイト数	11	2	512 バイト
クラスタサイズ	1 クラスタのセクタ数	13	1	64 セクタ
予約セクタ数	予約セクタ数（BPB を含む）	14	2	1 セクタ
FAT 数	FAT を何重に記録するか	16	1	2 個
rootDir サイズ	ディレクトリエントリ数	17	2	512 エントリ
総セクタ数 16	ボリュームのサイズ	19	2	0
FAT サイズ	FAT のセクタ数	22	2	245 セクタ
総セクタ数 32	ボリュームのサイズ	32	4	3,999,681 セクタ
ボリュームラベル	ボリュームの名前	43	11	`"MICRODRIVE␣"`
ブートプログラム	ブートプログラム	62	448	
シグネチャ	フォーマット済みマーク	510	2	`0x55 0xaa`

（位置と長さの単位はバイト）

（値の例はボリュームサイズ 2GiB，クラスタサイズ 32KiB，FAT16 の場合）

（「総セクタ数 16」で表現できない場合は「総セクタ数 32」を使用する）

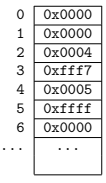
- 論理フォーマット時に決めたパラメータを記録している．
- 値の例は，ボリューム=2GiB，クラスタ=32KiB，FAT16
- ブートプログラムは初代 IBM PC の機械語で作成してある．

ディレクトリエントリ

Bytes	8	3	1	10	2	2	2	4
	FileName	Ext	Atr	Reserved	Time	Date	Cls	Size

- ルートディレクトリやディレクトリファイルに格納される．
- 前の BPB だとルートディレクトリに 512 個格納される．
- `FileName`：8 文字以内のファイル名
  （0x00：以降未使用，0xe5：削除，0x05：本当は 0xe5）
- `Ext`：3 文字以内の拡張子
- `Atr`：ファイルの属性
  （0x01：read-only，0x02：hidden，0x10：directory...）
- `Time`：ファイルの最終変更時刻
- `Date`：ファイルの最終変更日
- `Cls`：データが格納されている先頭クラスタの番号
- `Size`：ファイルの大きさ（バイト単位）

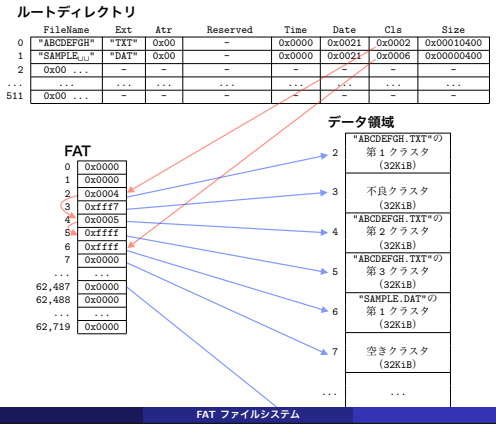
FAT（File Allocation Table）



値	意味
0x0000	未使用クラスタ
0x0001	予約クラスタ
0x0002 ～ 0xfff6	普通のクラスタ
0xfff7	不良クラスタ
0xfff8 ～ 0xffff	終了クラスタ

- クラスタをファイルに割り当てる表
- FAT エントリとクラスタが一対一対応
- FAT16 では FAT エントリが 16 ビット
- 0x0002～0xfff6 が普通のクラスタ番号
- **クラスタチェイン**を形成する．
- ディレクトリエントリの `Cls` がチェインの先頭を指す．

FAT ファイルシステムの全体像を示す例



ディレクトリファイル

```
$ cd /Volumes/NO\ NAME
$ mkdir A
$ mkdir A/DIR
$ echo AAA > A/A.TXT
$ hexdump -C A
00000000  2e 20 20 20 20 20 20 20  20 20 20 30 00 aa 7d 98  |.          0..}.|
00000010  e4 4c e4 4c 00 00 a9 98  e4 4c 21 00 00 00 00 00  |.L.L.....L!.....|
00000020  2e 2e 20 20 20 20 20 20  20 20 20 10 00 aa 7d 98  |..         ...}.|
00000030  e4 4c e4 4c 00 00 7d 98  e4 4c 00 00 00 00 00 00  |.L.L..}..L......|
00000040  44 49 52 20 20 20 20 20  20 20 20 10 00 31 a2 98  |DIR        ..1..|
00000050  e4 4c e4 4c 00 00 a2 98  e4 4c 31 00 00 00 00 00  |.L.L.....L1.....|
00000060  41 20 20 20 20 20 20 20  54 58 54 20 00 13 a9 98  |A       TXT ....|
00000070  e4 4c e4 4c 00 00 a9 98  e4 4c 40 00 04 00 00 00  |.L.L.....L@.....|
00000080  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
*
00008000
```

- ルートディレクトリ以外のディレクトリは `Atr=0x10` のファイル
- 上の実行例は macOS で FAT16 ファイルシステム上で実験したもの
- 最初の 2 エントリーは「.」と「..」を格納している．

練習問題

1. 次の言葉の意味を説明しなさい．
   - BPB
   - ルートディレクトリ
   - クラスタ
   - ディレクトリエントリ
   - FAT
   - クラスタチェイン
   - ディレクトリファイル
2. ディレクトリファイルのダンプリストをディレクトリエントリの構造と比較しながら解析しなさい．
3. 全体像を示す例の `ABCDEFGH.TXT` ファイルの第 0x00002000 バイトが格納されるクラスタの番号を答えなさい．
4. 前問と同様に，第 0x00004000，0x00008000，0x00010000 バイトに付いて答えなさい．