

Chương 2: Hệ Thống Files

Hệ điều hành

ThS. Đinh Xuân Trường
truongdx@ptit.edu.vn



Posts and Telecommunications
Institute of Technology
Faculty of Information Technology 1



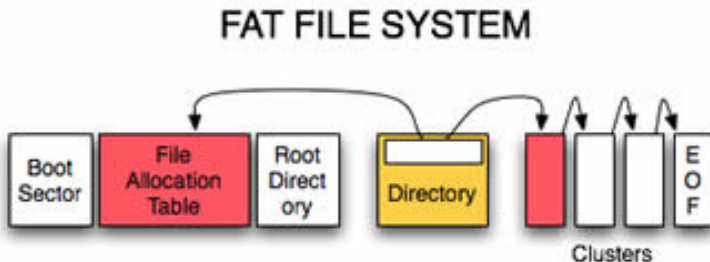
CNTT1
Học viện Công nghệ Bưu chính Viễn thông

August 15, 2022

Bài tập và thực hành

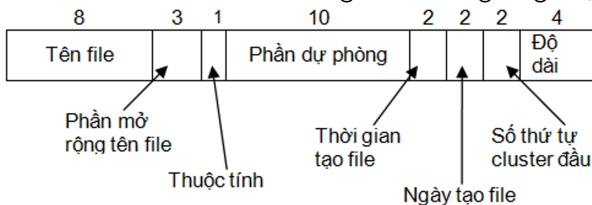
1. Các khái niệm
2. Các phương pháp truy cập file
3. Các thao tác với file
4. Thư mục
5. Cấp phát không gian cho file
6. Quản lý không gian trống trên đĩa
7. Độ tin cậy của hệ thống file
8. Bảo mật cho hệ thống file
9. Hệ thống file FAT

- ▶ Hệ thống FAT được thiết kế để sử dụng trong HDH DOS, sau đó đã được sử dụng trong một số phiên bản của HDH Window 3.0, 3.1, 95/98.



- ▶ Hiện nay, FAT vẫn là hệ thống thông dụng, được sử dụng trong hầu hết các HDH hiện nay để quản lý thẻ nhớ, đĩa mềm, đĩa CD
- ▶ 3 phiên bản: FAT12, FAT16, FAT32

- Chữ số chỉ kích thước ô bảng FAT tương ứng 12, 16 và 32 bit



- Hiện nay, FAT32 thường được sử dụng cho đĩa cứng, FAT16 được sử dụng cho thiết bị nhớ ngoài có dung lượng nhỏ như CD, thẻ nhớ ngoài.

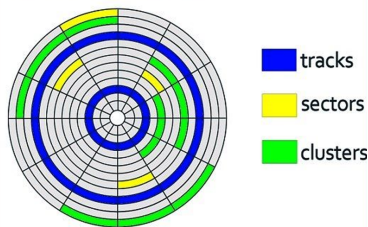
- ▶ Đơn vị tổ chức trong hệ thống file FAT là đĩa logic
- ▶ Đơn vị cấp phát không gian trên đĩa (khối logic) là các cluster.
- ▶ Mỗi cluster chứa một số lượng sector bằng lũy thừa của 2

Sector hay Cung là đơn vị nhỏ nhất do chương trình điều khiển đĩa cho phép đọc hoặc ghi (khối vật lý)

1 sector = 512byte

Cluster hay đơn vị cấp phát gồm một số khối vật lý, là đơn vị thông tin nhỏ nhất mà hệ điều hành cấp phát cho file (khối logic) 2KB-32KB

Hard disk drive structure



► Tổ chức thông tin trên đĩa logic:

Boot sector và các khối dự phòng	Bảng FAT1	Bảng FAT2	Thư mục gốc (chỉ có trên FAT12 và FAT16)	Phần còn lại cho tới cuối đĩa chứa các file và thư mục của đĩa lô gic
-------------------------------------	-----------	-----------	---	--

► Boot sector:

- Sector đầu tiên của đĩa logic
 - Chứa thông tin mô tả cấu trúc đĩa logic: kích thước sector, cluster, kích thước bảng FAT
 - Chứa mã chương trình khởi động để tải HDH nếu đĩa logic là đĩa khởi động
- **FAT:** bảng chỉ số quản lý cấp phát khối cho file. Do thông tin vùng này rất quan trọng nên HDH thường duy trì hai bảng FAT, trong đó một bảng là bản sao của bảng kia. Số lượng FAT có thể thay đổi và được ghi ở Boot sector.

- Tổ chức thông tin trên đĩa logic:

Boot sector và các khối dự phòng	Bảng FAT1	Bảng FAT2	Thư mục gốc (chỉ có trên FAT12 và FAT16)	Phần còn lại cho tới cuối đĩa chứa các file và thư mục của đĩa lô gic
-------------------------------------	-----------	-----------	---	--

- **Thư mục gốc ROOT:** vùng này chứa các thư mục gốc, thư mục trên cùng của đĩa. Kích thước vùng này bằng số lượng tối đa khoản mục trong thư mục gốc và được ghi trong boot sector. FAT32 không dùng vùng này mà lưu trữ thư mục gốc như các file trong thư mục thông thường.
- **Vùng dữ liệu:** chứa các file và thư mục của đĩa logic

Boot sector là sector đầu tiên trên đĩa và chứa thông tin mô tả cấu trúc đĩa cùng với mã để tải HDH đối với đĩa mềm HDH. Cấu trúc cụ thể của boot sector 32 Byte đầu tiên như sau:

Vị trí	Độ dài	Ý nghĩa
0	3	Lệnh Jump. Chỉ thị cho CPU bỏ qua phần thông tin và nhảy tới thực hiện phần mã khởi động của hệ điều hành nếu đây là đĩa mềm điều hành.
3	8	Tên hãng sản xuất, bổ sung dấu trắng ở cuối cho đủ 8B. Ví dụ: IBM 3.3, MSDOS5.0.v.v.
11	2	Bytes per sector. Kích thước sector tính bằng byte. Giá trị thường gặp là 512 đối với đĩa cứng. Đây cũng là vị trí bắt đầu của Khối Thông số BIOS (BIOS Parameter Block, viết tắt là BPB)
13	1	Sectors per cluster. Số sector trong một cluster, luôn là lũy thừa của 2 và không lớn hơn 128.
14	2	Reserved sectors. Số lượng sector dành cho vùng đầu đĩa đến trước FAT, bao gồm boot sector và các sector dự phòng.
16	1	Số lượng bảng FAT. Thường bằng 2.
17	2	Số khoản mục tối đa trong thư mục gốc ROOT. Chỉ sử dụng cho FAT12 và FAT16. Bằng 0 với FAT32.
19	2	Total sector. Tổng số sector trên đĩa. Nếu bằng không thì số lượng sector được ghi bằng 4 byte tại vị trí 0x20.
21	1	Mô tả loại đĩa. Ví dụ 0xF0 là đĩa mềm 3.5" hai mặt với 80 rãnh trên mỗi mặt, 0xF1 là đĩa cứng .v.v.
22	2	Sectors per FAT. Kích thước FAT tính bằng sector (đối với FAT12/16)
24	2	Sectors per track. Số sector trên một rãnh.
26	2	Number of heads. Số lượng đầu đọc (mặt đĩa được sử dụng)
28	4	Hidden sectors. Số lượng sector ẩn.
32	4	Total sector. Tổng số sector trên đĩa cho trường hợp có nhiều hơn 65535.

Các byte tiếp theo với FAT12/16:

Vị trí	Độ dài	Ý nghĩa
36	1	Số thứ tự vật lý của đĩa (0: đĩa mềm, 80h: đĩa cứng .v.v.)
37	1	Dự phòng
38	1	Dấu hiệu của phần mã mỗi. Chứa giá trị 0x29 (ký tự ')') hoặc 0x28.
39	4	Số xê ri của đĩa (Volume Serial Number) được tạo lúc format đĩa
43	11	Volume Label: Nhãn của đĩa được tạo khi format.
54	8	Tên hệ thống file FAT, ví dụ "FAT12 ", "FAT16 "
62	448	Mã mỗi hệ điều hành, đây là phần chương trình tải hệ điều hành khi khởi động.
510	2	Dấu hiệu Boot sector (0x55 0xAA)

Các byte tiếp theo với FAT32

Vị trí	Độ dài	Ý nghĩa
36	4	Sectors per FAT. Kích thước FAT tính bằng sector.
0x28	2	Cờ của FAT
0x2a	2	Version. Phiên bản.
0x2c	4	Số thứ tự của cluster đầu tiên của thư mục gốc root.
0x30	2	Số sector của Information Sector. Đây là phần nằm trong số sector dự phòng ngay sau boot sector.
0x32	2	Số thứ tự sector đầu tiên của bản sao của boot sector (nếu có)
0x34	12	Dự phòng
0x40	1	Số thứ tự vật lý của đĩa
0x41	1	Dự phòng
0x42	1	Dấu hiệu của phần mã mở rộng.
0x43	4	Số xê ri của đĩa (Volume Serial Number)
0x47	11	Volume Label
0x52	8	"FAT32"
0x5a	420	Mã môi hệ điều hành
0x1FE	2	Dấu hiệu Boot sector (0x55 0xAA)

- ▶ Quản lý các cluster trên đĩa và các file theo nguyên tắc:
 - Các khối thuộc cùng 1 file được liên kết thành 1 danh sách
 - Con trỏ được chứa trong ô tương ứng của bảng FAT
- ▶ Mỗi ô trong bảng FAT tương ứng với một cluster trên đĩa, chứa 1 trong các thông tin:
 - STT cluster tiếp theo trong danh sách các khối của file
 - Dấu hiệu kết thúc nếu ô tương ứng với cluster cuối cùng của file
 - Dấu hiệu đánh dấu cluster hỏng, không được sử dụng
 - Dấu hiệu đánh dấu cluster dự phòng
 - Bằng 0 nếu cluster trống, chưa cấp phát cho file nào

- ▶ Cluster đầu tiên của vùng dữ liệu được đánh STT là 2
- ▶ Hai ô đầu tiên của bảng FAT không dùng để quản lý cluster

FAT12	FAT16	FAT32	Ý nghĩa
0x000	0x0000	0x00000000	Cluster trống
0x001	0x0001	0x00000001	Cluster dự phòng, không được sử dụng
0x002–0xFEFF	0x0002–0xFFEF	0x00000002–0xFFFFFEFF	Cluster đã được cấp cho file. Chứa số thứ tự cluster tiếp theo của file.
0xFF0–0xFF6	0xFFF0–0xFFF6	0xFFFFFFFF0–0xFFFFFFFF6	Cluster dự phòng
0xFF7	0xFFFF7	0xFFFFFFFF7	Cluster hỏng.
0xFF8–0xFFFF	0xFFFF8–0xFFFFF	0xFFFFFFFF8–0xFFFFFFFFF	Cluster cuối cùng của file

Boot sector và các khối dự phòng	Bảng FAT1	Bảng FAT2	Thư mục gốc (chỉ có trên FAT12 và FAT16)	Phần còn lại cho tới cuối đĩa chứa các file và thư mục của đĩa lô gic
-------------------------------------	-----------	-----------	---	--

- ▶ Mỗi thư mục được lưu trong bảng thư mục, thực chất là 1 file đặc biệt chứa các khoản mục của thư mục
- ▶ Mỗi khoản mục chứa thông tin về một file hoặc thư mục con của thư mục đang xét
- ▶ Với FAT12/16, thư mục trên cùng của đĩa được chứa trong 1 vùng đặc biệt gọi là thư mục gốc
- ▶ Các thư mục mức thấp hơn/ thư mục gốc của FAT32 được chứa trong vùng dữ liệu trên đĩa cùng với các file

- Mỗi thư mục gồm các khoản mục 32 byte xếp liền nhau

Vị trí	Độ dài	Mô tả
0	8	Tên file, thêm bằng dấu trắng ở cuối nếu ngắn hơn 8 byte
8	3	Phần mở rộng, thêm bằng dấu trắng ở cuối nếu ngắn hơn 3 byte
11	1	Byte thuộc tính của file. Các bit của byte này nếu bằng 1 sẽ có ý nghĩa như sau:
		Bit 0: file chỉ được đọc; Bit 1: file ẩn; Bit 2: file hệ thống;
		Bit 3: Volume label; Bit 4: thư mục con Bit 5: archive; Bit 6: thiết bị nhớ khác (dùng cho hệ điều hành); Bit 7: không sử dụng
		Byte thuộc tính bằng 0x0F là dấu hiệu của file tên dài.
12	1	Dự phòng
13	1	Thời gian tạo file tính theo đơn vị 10ms, giá trị từ 0 đến 199
14	2	Thời gian tạo file theo format sau: bit 15-11: giờ (0-23); bit 10-5: phút (0-59); bit 4-0: giây/2 (0-29)
16	2	Ngày tạo file theo format sau. <i>Bit 15-9</i> : năm (0-1980, 127=2107); <i>bit 8-5</i> : tháng (1-12); <i>bit 4-0</i> : ngày (1-31)
18	2	Ngày truy cập cuối, theo format như ngày tạo file
20	2	2 byte cao của số thứ tự cluster đầu tiên của file trong FAT32
22	2	Thời gian sửa file lần cuối, theo format thời gian tạo file
24	2	Ngày sửa file lần cuối, theo format như ngày tạo file
26	2	Số thứ tự cluster đầu tiên của file trong FAT12/16.
28	4	Kích thước file tính bằng byte. Bằng 0 với thư mục con

► **int absread**(int *drive*, int *nsects*, long *lsect*, void **buffer*)

- *drive*: ổ đĩa cần đọc, A: 0, B:1, C:2
- *nsects*: số sector cần đọc
- *lsect*: vị trí sector bắt đầu đọc
- *buffer*: vùng nhớ lưu nội dung thông tin cần đọc

Sinh viên viết chương trình trên C/C++ để thực hiện các nội dung sau:

1. Đọc và in thông tin từ BOOT.
2. Đọc, phân tích, hiển thị nội dung bảng FAT.
3. Đọc, phân tích, hiển thị ROOT.
4. Duyệt số thứ tự hoặc nội dung các cluster của file cho trước.

Sinh viên viết chương trình trên $C/C++$ để thực hiện các nội dung sau:

1. **Bài 1:** Viết chương trình để hiển thị thông tin boot sector
2. **Bài 2:** Viết chương trình đọc FAT và in nội dung 100 ô FAT đầu tiên lên màn hình
3. **Bài 3:** Viết chương trình đọc ROOT và in nội dung giống lệnh DIR
4. **Bài 4:** Cho một tên file thuộc ROOT, viết chương trình tìm tất cả các cluster của file đó
5. **Bài 5:** Viết chương trình đến số cluster trống trong 100 cluster đầu tiên của ổ đĩa

Bài tập Thực hành hệ thống file FAT16:

1. **Bài 1:** Viết đoạn chương trình in nội dung của 150 ô FAT đầu tiên của ổ đĩa C ra màn hình
2. **Bài 2:** Giả sử bảng FAT được đọc vào bộ nhớ tại địa chỉ `<< int * fat >>` . Giả sử một file được lưu trữ trên cluster đầu tiên là *n*. Viết đoạn chương trình liệt kê các cluster thuộc về file đó.
3. **Bài 3:** Giả sử thư mục gốc của hệ thống file FAT16 sử dụng tên file độ dài tối đa 8 ký tự được đọc vào bộ nhớ tại địa chỉ `<< void * root >>`. Viết đoạn chương trình C/C++ thực hiện hai việc : in tên và độ dài các file trong thư mục gốc, tìm một file có tên cho trước trong thư mục gốc và số thứ tự cluster đầu tiên của file đó.

Chương 2

- ▶ Hệ thống file FAT
 - Đĩa logic
 - Boot sector
 - Bảng FAT
 - Root Thư mục gốc
 - Hàm đọc đĩa
 - Bài tập thực hành

Chương 3 Quản lý bộ nhớ

- ▶ Địa chỉ và các vấn đề liên quan
- ▶ Một số cách tổ chức chương trình
- ▶ Các yêu cầu quản lý bộ nhớ