Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
MỤC LỤC  
DANH MỤC HÌNH ẢNH 2  
TỔNG QUAN VỀ ĐỀ TÀI 3  
Chương 1. CƠ SỞ LÝ THUYẾT 4  
1.1. Tìm hiểu cấu trúc ổ đĩa cứng 4  
1.1.1. Cấu tạo vật lý của đĩa cứng 4  
1.1.2. Thông số và đặc tính 9  
1.2. Hệ thống FAT32 11  
1.2.1. Giới thiệu 11  
1.2.2. Cấu trúc FAT32 12  
1.2.3. Master Boot Record (MBR) 14  
1.2.3. Directory Table 15  
1.3. Hệ thống NTFS 17  
1.3.1. Giới thiệu 17  
1.3.2. Cấu trúc phân vùng NTFS 17  
1.4. So sánh giữa hệ thống FAT32 và NTFS 21  
Chương 2. THIẾT KẾ, XÂY DỰNG CHƯƠNG TRÌNH 24  
2.1. Phân tích yêu cầu 24  
2.2. Phân tích các chức năng 24  
2.3. Xây dựng các chức năng 25

Chương 3. TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ 30  
3.1. Môi trường triển khai 30  
3.2. Kết quả các chức năng của chương trình 30  
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 32  
PHỤ LỤC 34  
TÀI LIỆU THAM KHẢO 44  
Trương Xuân Hiếu – 11T1 - 1 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
DANH MỤC HÌNH ẢNH  
Trương Xuân Hiếu – 11T1 - 2 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
TỔNG QUAN VỀ ĐỀ TÀI  
1. Bối cảnh và lý do thực hiện đề tài  
Ổ đĩa cứng là một thiết bị rất quan trọng trong hệ thống bởi chúng chứa dữ liệu   
thành quả của một quá trình làm việc của những người sử dụng máy tính. Những sự hư   
hỏng của các thiết bị khác trong hệ thống máy tính có thể sửa chữa hoặc thay thế được,   
nhưng dữ liệu bị mất do yếu tố hư hỏng phần cứng của ổ đĩa cứng thường rất khó lấy   
lại được.  
Ngày nay, ổ đĩa cứng đã có những bước tiến công nghệ nhằm giúp lưu trữ và   
truy xuất dữ liệu nhanh hơn: ví dụ sự xuất hiện của các ổ đĩa cứng lai giúp cho hệ điều   
hành hoạt động tối ưu hơn, giảm thời gian khởi động của hệ thống, tiết kiệm năng   
lượng, sự thay đổi phương thức ghi dữ liệu trên các đĩa từ làm cho dung lượng mỗi ổ   
đĩa cứng tăng lên đáng kể.  
Để có thể cài đặt nhiều hệ điều hành đồng thời trên cùng một ổ đĩa cứng, buộc   
ta phải phân chia phân vùng giúp cho ổ đĩa cứng có thể định dạng các loại tập tin khác   
nhau, trong đó FAT32 và NTFS là hai định dạng cơ bản nhất của các ổ đĩa cứng hiện   
nay.  
Và để tìm hiểu kĩ hơn về nguyên tắc hoạt động của định dạng FAT32 cũng như   
NTFS, em quyết định chọn đề tài: Xây dựng chương trình đọc đĩa cứng với định dạng   
NTFS và FAT32. Qua đây, em cũng xin chân thành cảm ơn sự hướng dẫn của thầy   
Nguyễn Văn Nguyên đã giúp em hoàn thành đồ án này.  
2. Kết cấu của đồ án  
a. Tìm hiểu cấu trúc ổ đĩa cứng.  
b. Giới thiệu hệ thống FAT32, cấu trúc FAT32, partion Boot Sector, directory   
Table.  
c. Giới thiệu hệ thống NTFS, cấu trúc phân vùng NTFS, Master File Table.  
d. So sánh hệ thống FAT32 và NTFS.  
e. Xây dựng chương trình thực hiện các chức năng: Đưa ra màn hình Ổ đĩa, tên ổ đĩa,   
số serial, trạng thái, định dạng, dung lượng, còn trống, số byte/sector, số sector/cluster.  
Trương Xuân Hiếu – 11T1 - 3 -

Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
Chương 1. CƠ SỞ LÝ THUYẾT  
1.1. Tìm hiểu cấu trúc ổ đĩa cứng  
1.1.1. Cấu tạo vật lý của đĩa cứng  
Ổ đĩa cứng, hay còn gọi là ổ cứng (tiếng Anh: Hard Disk Drive, viết tắt: HDD)   
là thiết bị dùng để lưu trữ dữ liệu trên bề mặt các tấm đĩa hình tròn phủ vật liệu từ tính.  
Ổ đĩa cứng gồm các thành phần, bộ phận như sau:  
Hình 1. Cấu trúc ổ đĩa cứng  
1.1.1.1. Cụm đĩa  
Bao gồm toàn bộ các đĩa (gọi là đĩa từ), trục quay và động cơ.  
a. Đĩa từ (platter)  
Đĩa thường cấu tạo bằng nhôm hoặc thuỷ tinh, trên bề mặt được phủ một lớp   
vật liệu từ tính là nơi chứa dữ liệu. Tuỳ theo hãng sản xuất mà các đĩa này được sử   
dụng một hoặc cả hai mặt trên và dưới. Số lượng đĩa có thể nhiều hơn một, phụ thuộc   
vào dung lượng và công nghệ của mỗi hãng sản xuất khác nhau.  
Trương Xuân Hiếu – 11T1 - 4 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
Mỗi đĩa từ có thể sử dụng hai mặt, đĩa cứng có thể có nhiều đĩa từ, chúng gắn   
song song, quay đồng trục, cùng tốc độ với nhau khi hoạt động.  
Hình 2. Cấu tạo đĩa từ.  
Track: trên một mặt làm việc của đĩa từ chia ra nhiều vòng tròn đồng tâm gọi   
là các track. Track có thể được hiểu đơn giản giống các rãnh ghi dữ liệu như các đĩa   
nhựa (ghi nhạc trước đây) nhưng sự cách biệt của các rãnh ghi này không có các gờ   
phân biệt và chúng là các vòng tròn đồng tâm chứ không nối tiếp nhau thành dạng   
xoắn trôn ốc như đĩa nhựa. Track trên ổ đĩa cứng không cố định từ khi sản xuất, chúng   
có thể thay đổi vị trí khi định dạng cấp thấp ổ đĩa (low format). Thông thường mỗi đĩa   
từ có từ 312 đến 2048 track.  
Sector: Trên track chia thành những phần nhỏ bằng các đoạn hướng tâm gọi là   
các sector. Các sector là phần nhỏ cuối cùng được chia ra để chứa dữ liệu. Theo chuẩn   
thông thường thì một sector chứa dung lượng 512 byte. Mỗi track đều chia thành một   
lượng sector nhất định. Tuy nhiên, vì các track bên ngoài bao giờ cũng lớn hơn các   
track phía trong (gần trục) nên càng vào sâu các track phía trong thì dung lượng mà   
một sector có thể chứa được càng thấp.  
Cylinder: Tập hợp các track cùng bán kính (cùng số hiệu trên) ở các mặt đĩa   
khác nhau gọi là các cylinder. Nói một các chính xác hơn: khi đầu đọc/ghi đầu tiên   
làm việc tại một track nào thì tập hợp toàn bộ các track trên các bề mặt đĩa còn lại mà   
Trương Xuân Hiếu – 11T1 - 5 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
các đầu đọc còn lại đang làm việc tại đó gọi là cylinder. Trên một ổ đĩa cứng có thể có   
nhiều cylinder bởi có nhiều track trên mỗi mặt đĩa từ.  
Hình 3. Cấu trúc đĩa từ.  
b. Trục quay  
Trục quay là trục để gắn các đĩa từ lên nó, chúng được nối trực tiếp với động cơ   
quay đĩa cứng. Trục quay có nhiệm vụ truyền chuyển động quay từ động cơ đến các   
đĩa từ.  
Trục quay thường chế tạo bằng các vật liệu nhẹ (như hợp kim nhôm) và được   
chế tạo tuyệt đối chính xác để đảm bảo trọng tâm của chúng không được sai lệch - bởi   
chỉ một sự sai lệch nhỏ có thể gây nên sự rung lắc của toàn bộ đĩa cứng khi làm việc ở   
tốc độ cao, dẫn đến quá trình đọc/ghi không chính xác.  
Hình 4. Trục quay.  
Trương Xuân Hiếu – 11T1 - 6 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
c. Động cơ  
Được gắn đồng trục với trục quay và các đĩa. Chúng còn được gọi là động cơ   
bước - động cơ điện có nguyên lý và ứng dụng khác biệt với đa số các động cơ điện   
thông thường. Chúng làm việc nhờ các bộ chuyển mạch điện tử đưa các tín hiệu điều   
khiển vào stato theo thứ tự và một tần số nhất định.  
Chúng thực chất là một động cơ đồng bộ dùng để biến đổi các tín hiệu điều   
khiển dưới dạng các xung điện rời rạc kế tiếp nhau thành các chuyển động góc quay   
hoặc các chuyển động của rotour có khả năng cố định rotour vào các vị trí cần thiết.  
1.1.1.2. Cụm đầu đọc  
a. Đầu đọc (Head)  
Trên mỗi mặt đĩa từ của ổ cứng có một đầu đọc (head) riêng biệt, những đầu   
đọc này có vai trò đọc/ghi dữ liệu lên bề mặt đĩa từ.  
Đầu đọc đơn giản được cấu tạo gồm lõi ferit (trước đây là lõi sắt) và cuộn dây   
(giống như nam châm điện). Gần đây các công nghệ mới hơn giúp cho ổ đĩa cứng hoạt   
động với mật độ xít chặt hơn như: chuyển các hạt từ sắp xếp theo phương vuông góc   
với bề mặt đĩa nên các đầu đọc được thiết kế nhỏ gọn và phát triển theo các ứng dụng   
công nghệ mới.  
Hình 5. Đầu đọc (head).  
Đầu đọc trong đĩa cứng có công dụng đọc dữ liệu dưới dạng từ hóa trên bề mặt   
đĩa từ hoặc từ hóa lên các bề mặt đĩa khi ghi dữ liệu.  
Trương Xuân Hiếu – 11T1 - 7 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
Số đàu đọc ghi luôn bằng với số mặt hoạt động được của các đĩa cứng, có nghĩa   
chúng nhỏ hơn hoặc bằng hai lần số đĩa (nhỏ hơn trong trường hợp ví dụ hai đĩa nhưng   
chỉ sử dụng ba mặt).  
Ta có công thức:  
Dung lượng đĩa cứng = số head\*số cylinder\*số sector/track\*số mặt\*512byte.  
b. Cần di chuyển đầu đọc (head arm hoặc actuator arm):  
Cần di chuyển đầu đọc/ghi là các thiết bị mà đầu đọc/ghi gắn vào nó. Cần có   
nhiệm vụ di chuyển theo phương song song với các đĩa từ ở một khoảng cách nhất   
định, dịch chuyển và định vị chính xác đầu đọc tại các vị trí từ mép đĩa đến vùng phía   
trong của đĩa (phía trục quay).  
Các cần di chuyển đầu đọc được di chuyển đồng thời với nhau do chúng được   
gắn chung trên một trục quay (đồng trục), có nghĩa rằng khi việc đọc/ghi dữ liệu trên   
bề mặt (trên và dưới nếu là loại hai mặt) ở một vị trí nào thì chúng cũng hoạt động   
cùng vị trí tương ứng ở các bề mặt đĩa còn lại.  
Sự di chuyển cần có thể được thực hiện theo hai phương thức:  
- Sử dụng động cơ bước để truyền chuyển động.  
- Sử dụng cuộn cảm để di chuyển cân bằng lực từ.  
1.1.1.3. Cụm mạch điện  
a. Mạch điều khiển:  
Có nhiệm vụ điều khiển động cơ đồng trục, điều khiển sự di chuyển của cần di   
chuyển đầu đọc để đảm bảo đến đúng vị trí trên bề mặt đĩa.  
b. Mạch xử lý dữ liệu:  
Mạch xử lý dữ liệu dùng để xử lý những dữ liệu đọc/ghi của ổ đĩa cứng.  
c. Bộ nhớ đệm (cache hoặc buffer):  
Bộ nhớ đệm là nơi tạm lưu dữ liệu trong quá trình đọc/ghi dữ liệu. Dữ liệu trên   
bộ nhớ đệm sẽ mất đi khi ổ đĩa cứng ngừng được cấp điện.  
Trương Xuân Hiếu – 11T1 - 8 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
Bộ đệm của ổ cứng sử dụng một phần của RAM để lưu trữ những thông tin   
thường xuyên được các ứng dụng truy nhập. Chính việc lưu trữ những thông tin này   
trên RAM, bộ đệm đã giúp tốc độ truy xuất dữ liệu nhanh hơn và giúp kéo dài tuổi thọ   
của ổ cứng.  
d. Đầu cắm nguồn:  
Đầu cắm nguồn cung cấp điện cho ổ đĩa cứng.  
e. Đầu kết nối:  
Đầu kết nối giao tiếp với máy tính.  
f. Các cầu đấu thiết bị:  
Các cầu đấu thiết bị thiết đặt chế độ làm việc của ổ đĩa cứng: lựa chọn chế độ   
làm việc của ổ đĩa cứng (SATA 150 hoặc SATA 300) hay thứ tự trên các kênh giao   
tiếp IDE (master hay slave hoặc tự lựa chọn), lựa chọn các thông số làm việc khác…  
1.1.1.4. Vỏ đĩa cứng  
Vỏ ổ đĩa cứng gồm các phần: Phần đế chứa các linh kiện gắn trên nó, phần nắp   
đậy lại để bảo vệ các linh kiện bên trong.  
Vỏ ổ đĩa cứng có chức năng chính nhằm định vị các linh kiện và đảm bảo độ   
kín khít để không cho phép bụi được lọt vào bên trong của ổ đĩa cứng.  
Ngoài ra, vỏ đĩa cứng còn có tác dụng chịu đựng sự va chạm (ở mức độ thấp)   
để bảo vệ ổ đĩa cứng.  
1.1.2. Thông số và đặc tính  
1.1.2.1. Dung lượng  
Dung lượng ổ đĩa cứng (Disk capacity) là một thông số thường được người sử   
dụng nghĩ đến đầu tiên, là cơ sở cho việc so sánh, đầu tư và nâng cấp. người sử dụng   
luôn mong muốn ở hữu các ổ đĩa cứng có dung lượng lớn nhất có thể theo tầm chi phí   
của họ mà có thể không tính đến các thông số khác.  
Dung lượng ổ đĩa được tính bằng:  
(số byte/sector)\* (số sector/track)\* (số cylinder)\* (số head)  
Trương Xuân Hiếu – 11T1 - 9 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
Dung lượng của ổ đĩa cứng tính theo các đơn vị dung lượng cơ bản thông   
thường: byte, kB, MB, GB, TB.  
1.1.2.2. Tốc độ quay  
Tốc độ quay của đĩa cứng thường được ký hiệu là rpm (revolutions per minute):   
số vòng quay trong 1 phút. Tốc độ quay càng cao thì ổ cứng làm việc càng nhanh do   
chúng thực hiện việc đọc/ghi nhanh hơn, thời gian tìm kiếm thấp.  
Các tốc độ quay thông dụng thường là:  
- 3600 rpm: tốc độ của các ổ đĩa cứng thế hệ trước.  
- 4200 rpm: thường sử dụng với các máy tính xách tay mức giá thấp.  
- 5400 rpm: thông dụng với các ổ đĩa cứng 3.5”, sản xuất cách đây 8-9   
năm, và với các ổ đĩa cứng 2.5” cho các máy tính xách tay.  
- 7200 rpm: thông dụng với các ổ đĩa cứng sản xuất cách đây 2-3 năm.  
- 10000 rpm, 15000 rpm: thường sử dụng cho các ổ đĩa cứng trong các   
máy tính cá nhân cao cấp, máy trạm và các máy chủ có sử dụng giao tiếp   
SCSI.  
1.1.2.3. Bộ nhớ đệm  
Bộ nhớ đệm (cache hoặc buffer) trong ổ đĩa cứng cũng giống như RAM của   
máy tính, chúng có nhiệm vụ lưu tạm dữ liệu trong quá trình làm việc của ổ đĩa cứng.  
Độ lớn của bộ nhớ đệm có ảnh hưởng đáng kể tới hiệu suất hoạt động của ổ đĩa   
cứng bởi việc đọc/ghi không xảy ra tức thời (do phụ thuộc vào sự di chuyển của đầu   
đọc/ghi, dữ liệu được truyền tới hoặc đi) sẽ được đặt tạm trong bộ nhớ đệm.  
Đơn vị thường tính bằng kB hoặc MB.  
Hệ điều hành cũng có thể lấy một phần bộ nhớ của hệ thống (RAM) để tạo ra   
một bộ nhớ đệm lưu trữ dữ liệu được lấy từ ổ đĩa cứng nhằm tối ưu việc xử lý đối với   
các dữ liệu thường xuyên truy cập, đây chỉ là 1 các dùng riêng của hệ điều hành mà   
chúng không ảnh hưởng đến cách hoạt động hoặc hiệu suất vốn có của mỗi loại ổ đĩa   
cứng. Có rất nhiều phần mềm cho phép tinh chỉnh các thông số này của hệ điều hành   
tùy thuộc vào sự dư thừa RAM trên hệ thống.  
Trương Xuân Hiếu – 11T1 - 10 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
1.1.2.4. Chuẩn giao tiếp  
Các chuẩn giao tiếp của ổ đĩa cứng  
Giao tiếp (viết tắt) Tên tiếng anh đầy đủ Tốc độ truyền dữ liệu  
SCSI Small computer system   
interface  
Ultra160 SCSI 160 MBps  
Ultra320 SCSI 320 MBps  
ATA Advanced Technology   
Attachment  
Max = 133 MBps  
SATA 150 Serial ATA 150 150 MBps  
SATA 300 Serial ATA 300 300 MBps  
SATA 600 Serial ATA 600 600 MBps  
Trước đây, các chuẩn ATA và SATA thế hệ đầu tiên được sử dụng phổ biến   
trong máy tính cá nhân thông thường trong khi chuẩn SCSI và Fibe Channel có tốc độ   
cao hơn được sử dụng chủ yếu nhiều trong máy chủ và máy trạm. gần đây, các chuẩn   
SATA thế hệ tiếp theo với tốc độ giao tiếp cao hơn đang được sử dụng rộng rãi trong   
các máy tính cá nhân sử dụng các thế hệ chipset mới.  
1.2. Hệ thống FAT32  
1.2.1. Giới thiệu  
Được giới thiệu trong phiên bản Windows 95 Service Pack 2 (OSR 2), được   
xem là phiên bản mở rộng của FAT16. Do sử dụng không gian địa chỉ 32 bit nên   
FAT32 hỗ trợ nhiều cluster trên một partition hơn, do vậy không gian đĩa cứng được   
tận dụng nhiều hơn. Ngoài ra với khả năng hỗ trợ kích thước của phân vùng từ 2GB   
lên 2000GB và chiều dài tối đa của tên tập tin được mở rộng đến 255 ký tự đã làm cho   
FAT16 nhanh chóng bị lãng quên. Tuy nhiên, nhược điểm của FAT32 là tính bảo mật   
và khả năng chịu lỗi (Fault Tolerance) không cao.  
Trương Xuân Hiếu – 11T1 - 11 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
1.2.2. Cấu trúc FAT32  
Partition   
Boot Sector  
FAT1  
FAT2   
(duplicate)  
Root   
folder  
Other folders and all files  
1.2.2.1. Partition Boot Sector  
Chứa bảng tham số đĩa bao gồm thông tin về cấu hình đĩa, kích thước,… và loại   
hệ điều hành được cài đặt. Mã lệnh khởi động mồi bắt đầu cho hệ điều hành cũng được   
lưu ở đây.  
Đây là một sector đặc biệt nằm ở đầu mỗi partition đĩa. Đây là nơi mà các boot   
virus sẽ hiệu chỉnh lại nội dung. Để cấm việc hiệu chỉnh sector này bởi các ứng dụng   
(chủ yếu là virus), thường BIOS của các máy đời mới đều có chức năng bảo vệ boot   
sector, bất kỳ ứng dụng nào muốn hiệu chỉnh nội dung đĩa đều phải nhờ BIOS làm và   
BIOS sẽ kiểm tra, nếu sector bị hiệu chỉnh là boot sector thì nó sẽ hiển thị thông báo à   
bạn đã trình bày để người dùng viết và quyết định. Bạn có thể cho phép/cấm chức   
năng bảo vệ này của BIOS bằng cách vào BIOS Setup rồi thay đổi theo yêu cầu.  
Hình 6. Nội dung của một Boot Sector  
Máy tính dùng boot sector để chạy những chỉ dẫn trong suốt quá trình khởi động.   
Quá trình đó được tóm tắt như sau:  
- BIOS và CPU sẽ nạp power-on self test (POST).  
- BIOS sẽ tìm một boot device.  
Trương Xuân Hiếu – 11T1 - 12 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
- BIOS nập physical sector đầu tiên của boot device vào bộ nhớ và chuyển quyền   
điều khiển cho sector này.  
- Nếu boot device là HDD, BIOS sẽ nạp Master Boot Record (MBR). MBR code   
của MBR sẽ nạp boot sector của active partition và chuyển quyền điều khiển   
cho sector này. Trong Window 2000, executable code của boot sector sẽ tìm và   
nạp NTLDR vào memory và chuyển quyền thực thi cho file đó.  
- Nếu có một floppy disk trong ổ A, system BIOS nạp sector đầu tiên (the boot   
sector) của đĩa vào trong memory. Nếu như floppy disk là đĩa có thể khởi động,   
được format DOS với những file hệ thống, boot sector được load vào memory   
và dùng executable code để giao quyền thực thi cho file lo.sys - là một file hệ   
thống chính của MS-DOS.  
- Sau khi nạp OS vào memory, lúc này OS sẽ điều khiển hệ thống.  
1.2.2.2. Bảng FAT  
Bảng FAT nằm trong ngay sau cung khởi động (Volume Boot Sector). Mỗi   
Volume thường có hai bảng FAT 1 và FAT 2, nội dung 2 bảng này giống hệt nhau.  
FAT1, FAT2 là các bảng cấp phát và định vị file, thông tin chỉ mục giúp hệ   
điều hành có thể truy xuất chính xác đến file. Đồng thời qua bảng thông tin này hệ   
điều hành cũng xác định được dung lượng còn trống trên đĩa hoặc đánh dấu các vị trí   
BAD trên đĩa.  
Bảng FAT dùng để quản lý các Cluster. Bảng FAT chứa các lối vào (entry).   
Kích thước của 1 lối vào có thể là 12, 16 hoặc 32 bit, tùy thuộc vào kích thước của   
volume và cách tổ chức hệ tệp. Hai lối vào đầu tiên trong bảng FAT để chỉ ra dạng tổ   
chức của đĩa. Số lượng lối vào còn lại đúng bằng số lượng các Cluster. Các lối vào này   
cũng được đánh số thứ tự bắt đầu từ 2 và tăng dần đến tận lối vào cuối cùng trong   
bảng FAT. Mỗi 1 lối vào, bắt đầu từ lối vào số 2, chứa thông tin về trạng thái của   
Cluster có số thự tự tương ứng. Bảng FAT đóng vai trò một bản đồ về trạng thái các   
Cluster.  
Bảng FAT là sự ánh xạ của toàn bộ các cluster trên ổ đĩa, tuy nhiên FAT chỉ lưu   
thông tin về vị trí các cluster trên ổ cứng mà không lưu dữ liệu.  
Trương Xuân Hiếu – 11T1 - 13 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
1.2.2.3. Root Folder  
Bảng thư mục gốc giống như bảng thư mục của một cuốn sách mà chúng ta vẫn   
đọc, lưu trữ thông tin liên quan đến file hoặc thư mục như tên, ngày giờ tạo lập, thuộc   
tính file hoặc thư mục.  
1.2.2.4. Other file or folder  
Nơi lưu trữ thông tin thực sự của các file hoặc các thư mục con.  
1.2.3. Master Boot Record (MBR)  
Hình 7. Cấu trúc của MBR  
MBR là một mảng dữ liệu về các partition trong máy. Nó được tạo do hệ điều   
hành và dĩ nhiên mỗi hệ điều hành đều có cách tạo riêng biệt. Ví dụ, WinNT4 và   
Win2k sẽ có file boot.ini dùng như MBR. Mỗi khi khởi động máy, một chương trình   
chuyên làm việc loading hệ điều hành của WinNT và NT bootloader (một file loại.exe)   
sẽ dò tìm nội dung của file boot.ini để load hệ điều hành cần thiết cho người dùng.  
Riêng Windows 98 và Windows ME thì có chút khác biệt, chúng không có   
chương trình tìm hệ điều hành như NT bootloader và sẽ mặc nhiên load hệ điều hành   
tại phần partition đầu tiên trong ổ đĩa cứng, thông thường là partition C trong ổ đĩa   
cứng.  
Theo mặc định thì MBR sẽ luôn được lưu giữ tại ổ đĩa cứng thấp nhất (nếu ta   
có nhiều ổ đĩa cứng thứ tự là 0,1,2,…) và partition nhỏ nhất của nó (partition đầu tiên   
Trương Xuân Hiếu – 11T1 - 14 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
trong ổ đĩa số 0). Đây là một nơi rất quan trọng để BIOS tìm đến khi máy vừa khởi   
động.  
Trên MBR ngoài phần đoạn chương trình mồi khởi động thì còn có 4 vùng   
chứa tham số của logic chính (primary partition). Thông tin của vùng này giúp máy   
tình có thể xác định được ổ cứng được chia thành bao nhiêu ổ, ở vị trí nào và kích   
thước là bao nhiêu. Đây được gọi là partition table.  
1.2.3. Directory Table  
Một bảng thư mục là nơi chứa tất cả các chỉ mục về tập tin và thư mục. Mỗi file   
hoặc thư mục lưu trữ được đại diện bởi một chỉ mục dài 32 byte trong bảng. Mỗi chỉ   
mục lưu trữ thông tin về tên,đuôi,thuộc tính (nén,thư mục,ẩn,chỉ đọc,hệ thống và phân   
vùng),ngày và thời gian được tạo ra,địa chỉ của cluster đầu tiên của dữ liệu của file/thư   
mục và cuối cùng là kích thước của file/thư mục.   
Trừ bảng thư mục gốc trong FAT12 và file hệ thống FAT16 vốn chiếm một vị   
trí đặc biệt tại Root Directory Region,còn lại tất cả bảng thư mục được lưu trữ trên   
miền dữ liệu   
Byte   
Offset  
Chiều   
dài  
Mô tả  
0 8  
DOS file name (padded with spaces)   
The first byte can have the following special values:   
0x00 Entry is available and no subsequent entry is in use   
0x05 Initial character is actuall  
0x2E   
0xE5 'Dot' entry; either '.' or ' '   
0xE5 Entry has been previously erased and is now available   
Trương Xuân Hiếu – 11T1 - 15 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
8 3 DOS file extension (padded with spaces)   
11 1  
File Attributes   
The first byte can have the following special values:   
Bit Mask Description   
0 0x01 Read Only   
1 0x02 Hidden   
2 0x04 System   
3 0x08 Volume Label   
4 0x10 Subdirectory   
5 0x20 Archive   
6 0x40 Unused   
7 0x80 Unused   
An attribute value of 0x0F is used to designate a long file name   
entry.   
12 1 Reserved, used by NT   
13 1 Fine resolution creation time stamp, in tenths of a second   
14 4 Time of Creation   
18 2 Last Access Time   
20 2  
EA-Index (used by OS/2), High 2 bytes of first cluster number in   
FAT32   
22 4 Last Modified Time   
Trương Xuân Hiếu – 11T1 - 16 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
26 2  
First cluster in FAT12 and FAT16. Low 2 bytes of first cluster in   
FAT32.   
28 4 File size   
Bắt đầu từ WinNT 3.5, file được tạo trên đĩa FAT dùng các bit thuộc tính để hỗ   
trợ tên file dài mà không ảnh hưởng đến các hệ điều hành trước (DOS). Khi tạo một   
file, nếu tên file dài thì Windows sẽ tạo một tên dạng 8.3 cho file và sẽ thêm các điểm   
vào thứ cấp của file, mỗi điểm vào chứa 13 ký tự. DOS sẽ bỏ qua các điểm vào này,   
xem như chúng không tồn tại và xử lý như dạng file 8.3 chuẩn.  
1.3. Hệ thống NTFS  
1.3.1. Giới thiệu  
Hệ thống file NTFS (New Technology File System) được tạo ra vào những năm   
cuối 1990. Microsoft muốn tạo ra một sản phẩm công nghệ cao đáng tin cậy và bảo   
đảm cho hệ điều hành. Một trong những điểm yếu nhất của MS-DOS và Windows 3.x   
là chúng dựa vào hệ thống file FAT. Ngăn ngừa sự tàn lụi của Windows NT,   
Microsoft đã tạo ra một hệ thống file mới không dựa trên nền tảng hệ thống FAT. Kết   
quả là NTFS - hệ thống file công nghệ mới - ra đời. NTFS có quan điểm mới rạch ròi   
không dựa trên nền tảng hệ thống FAT cũ.   
Trong suốt quá trình phát triển của mình, từ hệ điều hành Windows NT,   
Windows 2000, Windows XP đến Windows Server 2003, NTFS đã có nhiều cải tiến   
để phù hợp với dung lượng lưu trữ lớn và yêu cầu bảo mật ngày càng cao.  
1.3.2. Cấu trúc phân vùng NTFS  
Partition   
Boot   
Sector  
Master File Table  
System   
files  
File area  
1.3.2.1. Partition boot sector  
Trương Xuân Hiếu – 11T1 - 17 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
Bảng boot sector trên ổ đĩa NTFS được mô tả khi ổ đĩa được định dạng NTFS.   
Khi bạn định dạng ổ đĩa NTFS, thì chương trình Format định 16 sector đầu tiên cho   
boot sector và phần chứa lệnh thực thi.  
Vị trí Độ dài Tên  
0x00 3 bytes Lệnh nhảy  
0x03 8 bytes OEM ID  
0x0B 25 bytes BPB  
0x24 48 bytes Dành cho phần mở rộng  
0x54 426 bytes Chứa lệnh thực thi  
0x01FE 2 bytes Tín hiệu kết thúc  
Trên ổ đĩa NTFS, trường dữ liệu được đánh theo trình tự từ BPB của phần mở   
rộng. Dữ liệu ở trường Ntdlr được kích hoạt trong suốt quá trình khởi động, trên ổ đĩa   
NTFS thì MFT không được đặt ở một sector xác định trước. Đó là lý do tại sao NTFS   
có thể di chuyển MFT nếu sector chứa MFT bị hỏng.  
1.3.2.2. Master File Table (MFT)  
Mỗi file trong định dạng NTFS được trình bày bằng 1 bản ghi trong 1 file đặc   
biệt gọi là Master File Table.  
NTFS cung cấp 16 bảng ghi đầu tiên trong bảng dành cho lưu trữ các thông tin   
đặc biệt. bảng ghi đầu tiên trong đó sẽ diễn ra chính bản thân MFT và được kèm theo 1   
bản sao của MFT.  
Vị trí của các đoạn dữ liệu của cả MFT và MFT mirror file đều được ghi trong   
boot sector. 1 bản sao của boot sector lại nằm trung tâm logic của đĩa.  
Trương Xuân Hiếu – 11T1 - 18 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
Hình 8. Cấu trúc MFT  
Standart   
information  
File or directory   
name  
Security   
descriptor  
Data or index  
Các file và thư mục nhỏ (cỡ 1500 byte hoặc nhỏ hơn) có thể được lưu trữ ngay   
trong bản ghi MFT. Cách sắp xếp này làm tốc độ truy cập trở nên nhanh hơn  
1.3.2.3. File Metadata  
Bởi vì MFT lưu trữ những thông tin về chính nó, NTFS dành riêng 16 record   
đầu tiên của MFT cho Metadata (xấp xỉ 16 KB), nó được dùng để mô tả cho MFT. File   
Metadata là những file mà chúng ta thấy bắt đầu tên là ký tự Dollar ($), chúng được   
mô tả ở bảng dưới đây. Các record còn lại dùng để chứa những file và thư mục cho   
mỗi thư mục hay tập tin trên đĩa.  
System File File Name MFT Record Purpose of the File  
Master file   
table  
$Mft 0  
Chứa đựng những file và thư mục cơ   
bản trên ổ đĩa NTFS  
Trương Xuân Hiếu – 11T1 - 19 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
Master file   
table mirror  
$MftMirr 1  
Đảm bảo cho sự truy cập tới MFT   
nếu trong trường hợp bị lỗi.  
Log file $LogFile 2  
Chứa đựng những thông tin của   
NTFS cho quá trình khôi phục nhanh   
hơn.  
Volume $Volume 3  
Chứng đựng những thông tin nói về ổ   
đĩa như nhãn đĩa,…  
Attribute   
definitions  
$AttrDef 4  
Danh sách những thuộc tính, những   
con số,…  
Root file name   
index  
. 5 Thư mục gốc.  
Cluster bitmap $Bitmap 6  
Mô tả ổ đĩa bằng hình ảnh mà không   
dùng tới Cluster.  
Boot sector $Boot 7  
Bao gồm BPB sử dụng để lắp đặt ổ   
đĩa và thêm vào lệnh thực thi.  
Bad cluster file $BadClus 8  
Chứa đựng những Cluster hỏng của ổ   
đĩa.  
Security file $Secure 9 Chứa đựng những mô tả về bảo mật.  
Upcase table $Upcase 10  
Chuyển đổi sang chữ hoa trong   
Unicode.  
NTFS   
extension file  
$Extend 11 Dùng để sử dụng nâng cấp sau này.  
1.3.2.4. File system data  
Lưu trữ thông tin, những thứ không chứa trong Master File Table.  
1.3.2.5. Master File Table Copy  
Bao gồm việc copy những thông tin cần thiết cho việc phục hồi dữ liệu, nếu file   
hệ thống có vấn đề gì thì nó sẽ đưa ra bản đã copy này để phục hồi lại dữ liệu.  
Trương Xuân Hiếu – 11T1 - 20 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
1.4. So sánh giữa hệ thống FAT32 và NTFS  
Bảng so sánh tính năng của NTFS, FAT32, FAT16  
Tính năng NTFS FAT32  
Hệ điều hành Windows NT/2000/XP  
Windows 98/ME/   
2000/XP  
Giới hạn  
Kích thước tối đa của phân   
vùng  
2 TB 2 TB  
Số tập tin tối đa trên một   
phân vùng  
Không giới hạn  
Không giới hạn  
Kích thước tối đa trên một   
phân vùng  
Giới hạn bởi kích thước   
phân vùng  
4 GB  
Số Cluster tối đa Không giới hạn 268.435.456  
Chiều dài tối đa của tên tập   
tin  
255 255  
Tính năng đặc trưng  
Hỗ trợ Unicode Có Có  
Systemrecord Mirror Tập tin ảnh MFT Bảng FAT thứ hai  
Vị trí Boot sector Đầu và cuối Chuẩn  
Thuộc tính của file Chuẩn hoặc tự thiết lập Chuẩn  
Khả năng nén Có Không  
Hiệu năng tổng quát  
Tính bảo mật Có Không  
Khả năng phục hồi Có Không  
Tận dụng không gian đĩa Cao Thấp  
Khả năng chịu lỗi Cao Thấp  
Trương Xuân Hiếu – 11T1 - 21 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
NTFS là hệ thống file tiên tiến hơn rất nhiều so với FAT32. Nó có đầy đủ các   
đặc tính của hệ thống file hiện đại mà FAT32 không hề có. Bạn nên dùng NTFS để   
thay thế cho FAT32 vì các lý do sau:  
- FAT32 không hỗ trợ các tính năng bảo mật như phân quyền quản lý, mã hóa,…   
như NTFS. Vấn đề này đặc biệt hiệu quả đối với Windows. Với NTFS, bạn có   
thể không cần sử dụng các tiện ích mã hóa hay đặt mật khẩu giấu thư mục… vì   
đây là đặc tính đã có sẵn của NTFS. Việc sử dụng các tiện ích không nằm sẵn   
trong hệ điều hành để thao tác trực tiếp với đĩa vẫn có ít nhiều rủi ro.  
- FAT32 có khả năng phục hồi và chịu lỗi rất kém so với NTFS. Có một số ý   
kiến cho rằng NTFS không tương thích nhiều với các chương trình kiểm tra đĩa   
hay sửa đĩa mà người dùng đã quen thuộc từ lâu, như vậy sẽ vô cùng bất tiện   
trong trường hợp đĩa bị hỏng sector. Người dùng nên yên tâm vì NTFS là hệ   
thống file có khả năng ghi lại được các hoạt động mà hệ điều hành đã và đang   
thao tác trên dữ liệu. Nó có khả năng xác định được ngay những file bị sự cố   
mà không cần phải quét lại toàn bộ hệ thống file, giúp quá trình phục hồi dữ   
liệu trở nên tin cậy và nhanh chóng hơn. Đây là ưu điểm mà FAT32 hoàn toàn   
không có.  
- Nếu mất điện đột ngột thì Windows 98, 2000, XP,… đều phải quét lại đĩa khi   
khởi động lại nếu đĩa đó được format bằng chuẩn FAT32. Trong khi format đĩa   
cứng bằng NTFS thì lại hoàn toàn không cần quét đĩa lại, bởi vì hệ thống dùng   
NTFS có được những thông tin về tính toàn vẹn dữ liệu ghi trên đĩa và nó mất   
rất ít thời gian để biết được về mặt logic đĩa của mình có lỗi hay không và nếu   
có thì hệ thống cũng tự phục hồi một các cực kỳ đơn giản và nhanh chóng. Với   
FAT32 thì nó phải rà quét toàn bộ lâu hơn nhiều. Một hệ thống Windows 2000,   
XP sẽ ổn định hơn nhiều nếu cài trên phân vùng được format bằng NTFS.   
Ngoài ra NTFS còn được trang bị công cụ kiểm tra và sửa đĩa rất tốt của   
Microsoft.  
- NTFS có khả năng truy cập và xử lý file nén tốt như truy cập vào các file chưa   
nén, điều này không chỉ tiết kiệm được dung lượng mà còn gia tăng được tuổi   
thọ của đĩa cứng.  
Trương Xuân Hiếu – 11T1 - 22 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
- Nhiều người phàn nàn rằng không thể truy cập vào các đĩa cứng được format   
bằng NTFS khi đang ở DOS, Windows 98 hoặc WinME,… Thực ra thì DOS,   
Windows 98 và Windows ME đã quá cũ và các phần mềm còn hữu dụng của   
chúng cũng không còn bao nhiêu.  
- NTFS được quota sử dụng cho người dùng, tiện dụng cho các hệ thống máy ở   
công ty, đặc biệt khi “âm thầm” cấm được con cái sao chép những phim ảnh   
độc hại vào các thư mục “bí mật” của chúng trong đĩa cứng.  
- Ngoài ra, NTFS còn có rất nhiều tiện ích chuyên sâu khác cho giới người dùng   
cao cấp khác như “mount partition”, tạo “hard link” tới một file, hỗ trợ dùng   
RAID,…  
- Nếu bạn đã thực sự quyết định chọn NTFS thì bạn có thể từ bỏ hẳn FAT32 kể   
từ nay. Hiện có rất nhiều tiện ích chuyển đổi từ FAT 32 sang NTFS tùy bạn lựa   
chọn. Tiện hơn cả là dùng bộ tiện ích có sẵn trong các đĩa CD khởi động bằng   
Hirenboot đang rất phổ biến hiện nay.  
- Tuy thế, FAT32 vẫn còn tỏ ra hữu dụng trên các máy tính cấu hình quá yếu ớt,   
chỉ có thể chạy được Windows 98. FAT16 và FAT32 vẫn được dùng để định   
dạng cho các loại thẻ nhớ, vì các thiết bị chấp nhận thẻ nhớ như máy ảnh số,   
máy nghe nhạc vẫn chưa thấy loại nào tương thích với NTFS cả. FAT16 luôn là   
lựa chọn hàng đầu khi bạn muốn copy dữ liệu của mình từ một máy tính chạy   
Windows sang máy chạy hệ điều hành khác như MAC chẳng hạn. Hầu hết các   
máy Macintosh hiện nay đều không thể nhận dạng các thẻ nhớ USB được định   
dạng bằng FAT32.  
Trương Xuân Hiếu – 11T1 - 23 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
Chương 2. THIẾT KẾ, XÂY DỰNG CHƯƠNG TRÌNH  
2.1. Phân tích yêu cầu  
Bài toán yêu cầu: Xây dựng chương trình thực hiện các chức năng: Đưa ra màn   
hình ổ đĩa, tên ổ đĩa số serial, trạng thái, định dạng, dung lượng, còn trống, số   
byte/sector, số sector/cluster.  
2.2. Phân tích các chức năng  
Xây dựng chương trình thực hiện các chức năng sau:   
- Đọc thông tin các logical partitions (phân vùng logic) của ổ cứng và hiển thị   
lên màn hình thông qua giao diện. Thông tin của partitions logical bao gồm:  
rootPathName : Đường dẫn.  
volumeName : Tên phân vùng logic (logical partition).  
filesystemName : Tên định dạng (NTFS,FAT32,…).  
sectorsPerCluster : Số sectors/Cluster.  
numberOfFreeClusters : Số Clusters chưa sử dụng (dung lượng còn trống của   
phân vùng logic).  
totalNumberOfClusters : Tổng số Clusters (Dung lượng phân vùng logic).  
volumeSerialNumber : Số hiệu phân vùng logical.  
type : Kiểu của phân vùng logical.  
- Đọc thông tin của HardDisk (ổ đĩa cứng) và hiển thị lên màn hình thông qua   
giao diện. Thông tin của HardDisk bao gồm:  
vendorId : Tên nhà sản xuất HardDisk.  
productId : Mã sản phẩm HardDisk.  
serialNumber : Số hiệu HardDisk.  
bytePerSector : Số Bytes/Sector (thông thường là 512 bytes).  
sectorsPerTrack : Số Sectors/Track.  
Trương Xuân Hiếu – 11T1 - 24 -  
Báo cáo ĐA Nguyên lý Hệ Điều Hành GVHD: Nguyễn Văn Nguyên  
tracksPerCylinder : Số Tracks/Cylinder.  
diskSize : Dung lượng HardDisk.  
cylinders : Số Cylinders có trong HardDisk.  
driverType : Kiểu HardDisk.  
2.3. Xây dựng các chức năng  
Ta xây dựng class DiskInfo để thực hiện các chức năng đã đề cập ở trên.  
Ta sẽ xây dựng cấu trúc LogicalPartitionInfo chứa các giá trị thông tin của   
Logical Partition và cấu trúc HardiskInfo chứa các giá trị thông tin của HardDisk.  
struct LogicalPartitionInfo  
{ string rootPathName, volumeName, fileSystemName;  
unsigned long sectorsPerCluster, numberOfFreeClusters,   
totalNumberOfClusters;  
unsigned long volumeSerialNumber;  
long long size;  
int type;  
};  
struct HardiskInfo   
{ string vendorId, productId, serialNumber;  
unsigned long bytePerSector, sectorsPerTrack, tracksPerCylinder;  
long long diskSize, cylinders;  
int driverType;   
};  
Tiếp theo định nghĩa 2 danh sách để lưu kết quả đọc được:  
typedef vector<LogicalPartitionInfo \*> ListLogicalPartitionInfo;  
typedef vector<HardiskInfo \*> ListHardiskInfo;  
Sau đó ta xây dựng các hàm thực hiện việc đọc và in thông tin Logical Partition   
và HardDisk:  
Trương Xuân Hiếu – 11T1 - 25 -