

교환 시스템에서의 이중화 저장장치

정회원 노 승 환*

Redundant Storage Device in Communication System

Soonghwan Ro* *Regular Member*

요 약

일반적으로 교환 시스템은 프로세서 보드, 입출력처리 보드 및 데이터 저장 장치등 그 기능별로 다수의 서브시스템들로 구성되어 있다. 또한 신뢰성을 확보하기 위하여 모든 서브 보드들은 이중화로 되어 있다. 교환시스템에서 저장 장치에는 시스템에 관련된 정보나 과금 정보등과 같은 작업 관련 데이터를 저장하며 비 휘발성 메모리에 저장해야 한다. 일반적으로 비 휘발성 저장장치를 구현할 때는 플래시메모리(flash memory) 또는 배터리 백업 메모리(battery backup memory)를 사용한다. 그러나 메모리는 단위 용량당 가격이 높고 백업(backup) 하지 않은 데이터를 손실했을 때 복구할 수 없다.

본 논문에서는 마이크로 컨트롤러를 이용하여 단위 용량당 가격이 저렴하고, 대용량의 데이터를 저장함과 동시에 이중화를 만족시키는 on-board 형태의 소형 디스크 모듈을 설계 구현하였다.

본 논문에서 구현된 이중화 저장장치는 사용 중인 액티브(active) 디스크에 결함이 생겨 사용할 수 없을 경우에 리빌딩(rebuilding)과정을 통해 스탠바이 모듈로부터 데이터를 복구하며, 리빌딩 중에도 호스트 시스템은 스탠바이 디스크모듈을 이용하여 지속적으로 서비스를 제공할 수 있도록 설계되었다.

본 연구에서 개발된 저장장치는 교환시스템에서 플래시 메모리와 같은 값비싼 저장 장치를 대체할 수 있을 것으로 기대된다.

키워드 : 교환시스템, 이중화 저장장치

ABSTRACT

In general communication system is composed of processor subsystems, I/O processor subsystems and data storage device subsystems those are classified as their functions. In order to improve the data reliability, all subsystems are redundant. Storage device keeps the operational information such as system related information and charging information, and such informations must be stored in non-volatile memory. Flash memory and battery backup memory are commonly used as the non-volatile storage devices. But such kind of memories are expensive per unit capacity and data can't be restored when lost while not being backed up.

In this paper we develop a redundant storage device to store a lot of data safely and reliably in communication system. The device consists of micro-controller, FPGA and hard disk. It provides many functions those are rebuilding, automatic remapping, host service and remote host service. Also it is designed to provide host service while rebuilding is being done in order not to interrupt the communication services.

The developed device can be used instead of expensive storage device like flash memory in various communication systems.

1. 서 론

일반적으로 통신시스템은 프로세서 보드, 입출력 처리용 보드 및 데이터 저장용 보드 등 그 기능별

로 다수의 보드(board)들로 구성되어 있으며, 신뢰성을 확보하기 위해 액티브(Active) 및 스탠바이(Standby)보드로 이중화되어 있다.

그러나 액티브보드의 장애로 인해 스탠바이보드

* 공주대학교 정보통신공학과(rosh@mail.kongju.ac.kr)

논문번호: 030286-0703, 접수일자: 2003년 7월 3일

로 절체가 이루어질 때 스텐바이보드는 현재 진행 중인 작업을 계속 할 수 있어야 하며, 이를 위하여 진행 중인 작업과 관련한 모든 데이터는 주 기억 장치에 저장되어야 하고, 그 외의 시스템 정보나 프로그램 정보등과 같은 작업 관련 데이터는 주로 비휘발성 저장장치에 존재해야 한다.

통신시스템에서 비휘발성 저장장치를 구현할 때에는 일반적으로 플래시메모리(Flash Memory), 배터리백업 메모리(Battery Backup Memory)등을 사용한다.

그러나 플래시메모리는 적은 공간을 차지하며 속도가 빠르다는 장점에 비해서 DRAM 보다 비싼 단가, 블록을 지우기 위한 시간 동안의 서비스 중단, 그리고 쓰기 횟수의 제한 등의 단점을 가지고 있다.

그리고 배터리 백업을 가진 RAM, 예를 들어 SRAM에 배터리를 부착한 NVRAM과 같은 제품은 그 최대 용량이 수 MB에 그치기 때문에 극히 제한적인 용도에만 사용할 수 있으며, 수십~수백 MB에 이르는 용량의 메모리를 구현하려면 비용이 과다하게 소요되고 공간상으로 문제가 발생하여 현실적으로 사용가능성이 희박하다. 이러한 이유로 기존의 플래시메모리나 배터리백업메모리를 대체할 비휘발성 메모리로 하드디스크를 사용가능하다.

본 논문에서는 마이크로컨트롤러(Micro-Controller)를 이용하여 대용량 저가가격의 비휘발성 메모리의 요구와 이중화 기능을 동시에 만족시키는 온보드(on-board) 형태의 디스크 모듈을 구현하고자 한다. 본 연구에서 구현된 디스크 모듈은 액티브 보드와 스텐바이 보드에 부착된 디스크 모듈은 동일한 데이터를 유지하고 있다가 액티브 디스크 모듈에서 에러가 발생했을 경우에 감지하여 해당섹터를

스텐바이 디스크 모듈에서 자동 복구할 수 있으며, 또한 새로운 디스크 모듈을 장착했을 경우에 스텐바이 보드의 디스크 모듈에서 전체 데이터를 복구할 수 있는 리빌딩(rebuilding) 기능을 가지며, 이는 일반 비휘발성 메모리를 사용한 시스템에서 개선된 기능들이다.

본 연구에서 사용된 디스크 이중화 저장장치는 RAID 시스템의 레벨 1 미러링(mirroring) 기능을 임베디드 디스크 모듈로 구현하여 교환시스템에 적용한 형태이다. 일반적으로 디스크 미러링 기능은 동일한 데이터를 두개의 디스크에 동시에 저장함으로써 데이터의 신뢰성을 높이기 위한 목적이 있다.[1][2]

II. 저장장치에 관한 고찰

2.1 비휘발성 저장장치간의 비교

비휘발성 저장장치로 사용할 수 있는 장치로는 플래시메모리와 소형 하드디스크외에 콤팩트플래시 메모리와 마이크로드라이브등을 사용할 수 있다. 이외에도 FRAM(Ferroelectric RAM)과 같은 다른 장치들이 사용할 수 있겠으나 아직 대중적으로 낮은 가격에 사용할 수 있을 만큼 널리 보급되지는 않은 상태이다.

표 1에는 사용가능한 저장장치간의 특성을 비교하였다.

위 장치들은 서로 비교하는 특성과 기준이 다를 수 있다. 특히 수명과 신뢰성을 비교할 때는 각 장치간의 평가 기준이 서로 다르므로 보는 관점에 따

표 1. 비휘발성 저장장치간의 특성비교

비교항목 \ 장 치	플래시 메모리 (Flash Memory)	컴팩트플래시 메모리 (CompactFlash Memory)	마이크로드라이브 (Micro Drive)	2.5" 하드디스크 (2.5" HDD)
비교장치	Intel strataflash memory(32MByte)	Transcend CompactFlash 2.2Gbyte	Hitachi Microdrive 1Gbyte	Hitachi Travelstar 40Gbyte
수 명	100,000 cycle	1,000,000 program/erase		
가 격(\$/16Mbyte)	76	1.28	2.375	0.05
전송속도(Mbyte/sec)	16.7	16.6	13.3	12.5
단위면적당 용량 (Mbyte/cm ²)	11.41	144.54	65.7	546

라 많은 차이를 보일 수가 있으며, 전송속도도 디스크 장치와 메모리 장치사이에 데이터에 접근하는 방식이 다르지만 본 비교에서는 최초의 데이터를 접근한 시점에서 연속적으로 데이터를 외부로 전송하는 전송율을 비교하였다. 용량비교에서는 저장 미디어를 둘러싸는 패키지 크기에 의해 많은 영향을 받을 수가 있으므로 본 비교표에서는 2.5" 하드디스크를 제외하고는 동일한 장치 중에서 가장 용량이 큰 모델을 대상으로 비교하였으며, 2.5" 하드디스크는 본 논문에서 사용된 모델을 대상으로 하였다.

위의 표에서 사용된 특성들은 각 제품에 대한 데이터 매뉴얼과 specification중 가장 최근의 자료를 기준으로 작성되었으며[3][4][5][6], 가격은 동일한 제품을 취급하는 외국의 인터넷 사이트(<http://shopper-zdnet.com.com>)를 기준으로 작성되었다.

위의 표에서 볼 수 있듯이 하드디스크는 단위 용량당 가격과 단위면적당 용량에서 가장 큰 장점을 가지고 있다. 그러나 단점으로는 플래시메모리와 비교했을 때 전송속도에서 낮아서 약 75%~80%의 전송속도를 보인다.

III. 이중화 저장장치의 기능

3.1 기능에 대한 정의

통신시스템에서 하드디스크를 이용하여 이중화 저장장치를 구현하기 위해서는 그림 1에서 나타내는 것과 같이 다음과 같은 기능을 제공해야 한다.

(1) 액티브 호스트 액세스(Active Host Access)

액티브 보드에서 이중화 저장장치에 데이터를 쓰기(write) 할 경우, 항상 같은 데이터를 유지하기 위해서 액티브보드뿐만 아니라 스탠바이보드의 이중화 저장장치에도 같은 데이터를 저장해야 하며, 읽기(read) 시에는 단일보드와 같이 액티브보드의 장치에서만 데이터를 읽는다.

(2) 스탠바이 호스트 액세스(Remote Access)

스탠바이보드에서의 읽기인 경우에는 항상 같은 데이터를 가지고 있으므로 액티브보드의 이중화 저장장치에서 데이터를 읽는다.

(3) 데이터 리빌딩(Data Rebuilding)

액티브보드에 설치된 시스템에 이중화를 위한 스

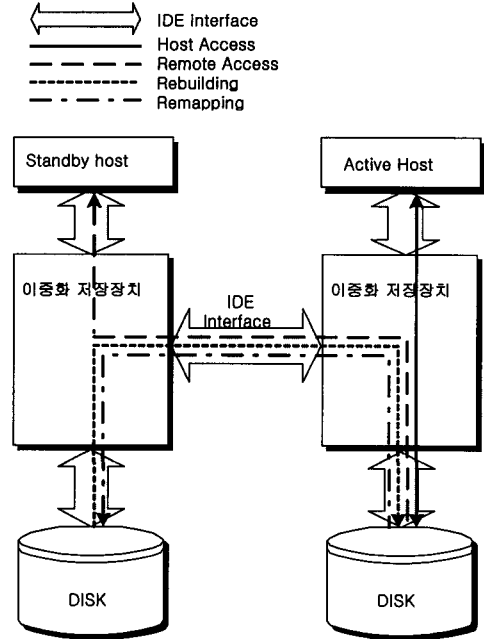


그림 1. 이중화 저장장치의 기능

탠바이보드의 이중화 저장장치가 추가로 설치되면, 이를 자동적으로 감지하여 액티브보드의 이중화 저장장치에 담겨있는 데이터를 액티브보드의 동작상태를 유지하면서 스탠바이보드의 이중화 저장장치에 복사해서 두 개의 저장장치가 항상 동일한 데이터를 갖도록 한다. 이때 소요되는 시간은 2GB기준으로 10분 이내로 하여 전송속도문제를 해결한다.

(4) 데이터 리맵핑(Data Remapping)

하드디스크의 하나의 섹터(Sector)에 결함(Defect)이 발생하여도 전체 디스크를 교체해야 한다면, 제품의 전체 수명이 상당히 짧아진다. 그래서 이를 해결하기 위하여 자동리맵핑 기능을 보유하고 있는 하드디스크를 사용하고, 이중화 저장장치에서 결함이 발생하면 이중화 되어있는 저장장치로부터 정상적인 데이터를 읽어 들여서 자동적으로 복구할 수 있도록 한다.

(5) 데이터 리빌딩 중 스탠바이보드에서 호스트 서비스 제공

데이터 리빌딩 시간이 약 10분정도 소요되는 동안, 스탠바이의 호스트에서 디스크 액세스를 하지 못하게 되면 부팅 시간이 그만큼 길어진다. 따라서 부팅 시간을 줄이기 위해서 스탠바이보드의 이중화

저장장치에서 데이터 리빌딩 중에도 액티브보드의 저장장치에서 디스크를 읽어내어서 호스트의 동작을 지원하여야 한다.

IV. 이중화 저장장치의 설계 및 구현

4.1 이중화 보드의 구성

그림 2 에서와 같이 이중화 보드의 구성은 크게 프로세서, FPGA, 하드디스크로 나뉘며 하드디스크와의 인터페이스는 IDE방식[7][8]으로 구성된다.

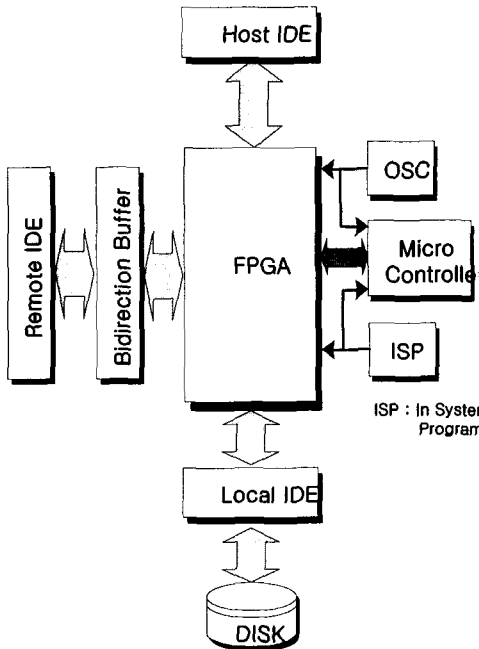


그림 2. 이중화 저장장치의 구성

사용된 마이크로컨트롤러(Micro-Controller)는 8051 컨트롤러와 호환성을 갖는 프로세서[9][10]를 사용하였으며, 내부메모리를 가지고 있으며, ISP (In-System Programming)가 가능하여 이중화 저장장치의 소형화와 프로그램 업그레이드(Program Upgrade)를 용이하게 하였다. 또한 FPGA도 ISP가 가능하게 함으로써 하드웨어 업그레이드가 가능하도록 하였다. IDE(Integrated Disk Electronics) 인터페이스는 하드디스크를 액세스하기 위하여 사용되는

인터페이스로서 데이터 전송은 512byte 단위이며, 빠른 데이터 전송을 위해 Multiple DMA 전송을 지원하고 프로세서와의 데이터전송은 PIO (Programmed I/O)로 전송한다. 또한 하드디스크(Hard Disk Drive)는 IDE 인터페이스 방식을 지원하며, 디스크에 데이터를 쓰기(write)시에 자동 리맵핑 기능을 가진다.

그 외 스탠바이 보드와 데이터를 주고받을 때 거리에 따른 IDE 컨트롤(control) 신호의 감쇠를 보상하기 위하여 양방향 버퍼(Bidirectional Buffer)를 사용하였으며, 프로세서와 FPGA의 ISP회로를 추가하였다.

4.2 각 블록의 설계 및 구현

그림 3 시스템 블록도는 마이크로컨트롤러와 FPGA를 이용하여 위의 기능들을 구현하기 위한 전체 블록도를 나타내며, 각 블록의 기능은 다음과 같다.

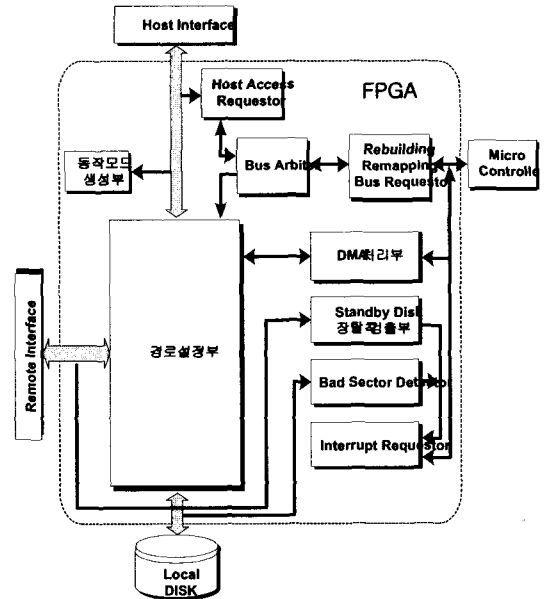


그림 3. 이중화 보드의 전체 시스템 블록도

(1) 마이크로 컨트롤러

컨트롤러의 외부 인터럽트(external interrupt)신호를 이용하여 인터럽트 요청부에서 리빌딩 및 리맵핑 신호를 받아서 인터럽트 서비스 루틴을 수행한다.

(2) 리빌딩, 리맵핑 버스 요구부

마이크로컨트롤러에서 버스요구가 들어오면 버스중재부(Arbiter)에 버스를 요청하고 허가신호(bus grant)를 기다려 결과를 마이크로컨트롤러에 알려준다.

(3) 버스중재부(Bus Arbiter)

리빌딩, 리맵핑, 호스트엑세스, 스탠바이 호스트 엑세스요청 신호가 들어오면 우선순위(리빌딩=리맵핑>호스트엑세스> 리모트 엑세스)에 맞게 동작모드를 결정하여 동작모드 생성부와 경로 설정부에 알려준다.

(4) 호스트 엑세스 요구부(Host Access Requester)

액티브 보드에서의 Read/Write신호를 받아 버스중재부에 서비스를 요청한다.

(5) 동작모드 생산부

버스중재부에서 동작모드를 입력받아 스탠바이 보드로 액티브 보드의 현재 진행상태를 알려준다.

(6) DMA 처리부(DMA Engine)

리빌딩, 리맵핑 과정을 수행시 빠른 데이터 전송을 위하여 Multiple DMA 방식으로 구현하였으며 리빌딩 수행시 호스트 서비스를 제공하기 위하여 전송할 섹터 수를 임의의 수로 나누어 이를 사용자가 조절할 수 있도록 하였다.

(7) 스탠바이 디스크 장탈착 검출부

액티브 보드가 단일 보드로 동작하다가 스탠바이 보드가 장착되는 시점부터 리빌딩이 진행되어야 하므로 스탠바이 보드의 이중화 저장장치가 연결되는 시점을 감지하여 인터럽트 요구부에 알려준다.

(8) 베드섹터 검출부(Bad Sector Detector)

베드섹터에 대하여 자동 리맵핑 기능을 제공하는 하드디스크 드라이브를 사용하게 되면, 쓰기 동작에 대해서는 에러가 발생하지 않으나, 읽기 동작에서는 에러가 발생할 수 있다. 여기서는 IDE인터페이스의 상태레지스터의 busy bit를 체크하여 에러를 감지한다.

(9) 인터럽트 요구부(Interrupt Requester)

베드섹터 검출시, 스탠바이 디스크 장탈착 검출시, DMA 전송시 발생하는 인터럽트를 처리한다.

표 2. 컨트롤러의 외부 인터럽트의 정의

외부 인터럽트	인터럽트 종류
EXT_INT0	Rebuilding Interrupt DMA Interrupt
EXT_INT1	Remapping Interrupt

(10) 경로설정부(Path Control)

버스중재부에서 동작모드를 입력받아 각 기능에 맞게 IDE 인터페이스의 컨트롤신호나 데이터버스경로를 설정하여준다.

4.3 소프트웨어 구성

소프트웨어는 외부 인터럽트가 발생할 때까지 IDLE 상태로 대기하고 있으며, 외부 인터럽트 0 혹은 외부 인터럽트 1 이 발생하면 IDLE 상태에서 벗어나 각 인터럽트에 알맞은 동작을 수행한다.

수행 동작으로는 리빌딩과 리맵핑이 있으며, 리빌딩의 경우 수행 중 호스트로부터의 디스크 엑세스 요청을 고려하여 몇 개 단위의 섹터로 DMA 전송을 수행한다. 반면 리맵핑은 한 개의 섹터에 대한 DMA 전송을 수행한다.

프로그램에서의 처리상태는 크게 다섯 상태로 구분되어 진다. 첫 번째로 IDLE Mode로 대기상태로 인터럽트를 기다리다가 인터럽트 요청 신호가 입력되면 인터럽트종류(Rebuilding, Remapping)를 확인하여 인터럽트 서비스 루틴을 호출한다. 두 번째와 세 번째로 Rebuilding Setup, Reset Mode와 Remapping Setup, Reset Mode로 Rebuilding과 Remapping을 수행하기 위하여 IDE Register를 초기화시킨다. 네 번째로 Rebuilding DMA Mode로 액티브보드의 디스크에서 스탠바이보드의 디스크로 데이터를 DMA 방식으로 리빌딩을 수행한다. 다섯 번째로 Remapping DMA Mode는 결함이 발생한 한 개의 섹터를 DMA 방식으로 리맵핑 시킨다.

그림 4는 위에서 설명한 진행모드의 동작 상태도를 나타내고 있다.

단, 리빌딩 과정 중에 발생하는 읽기 오류(read error) 및 리맵핑 중의 읽기 오류 경우, 해당 섹터는 베드 섹터로 간주하여 더 이상의 동작을 진행시키지 않는다. 리빌딩의 경우에는 다음 섹터부터 리빌딩을 진행하고, 리맵핑의 경우 리맵핑 동작을 종료한다. 리빌딩 과정 중에 쓰기 동작에서 에러가 발생하면 스탠바이보드의 디스크에서 에러가 발생한 것으로 간주하고 리맵핑 동작을 시도한다. 그러나 리

맵핑과정에서 또 에러가 발생하면 해당 섹터에 대해 리빌딩을 포기하고 다음 섹터부터 리빌딩을 수행한다.

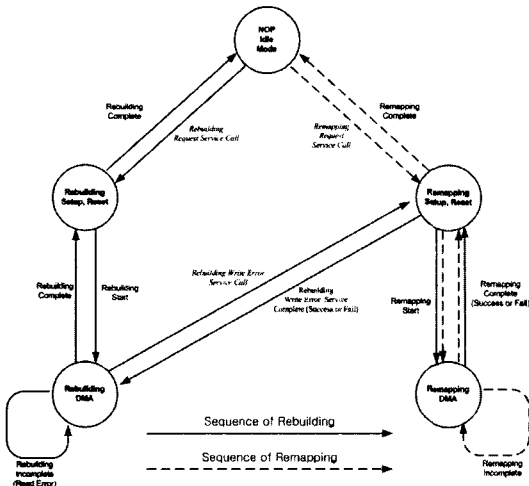


그림 4. 프로그램 동작 상태도

V. 구현 결과

5.1 데이터 전송 시간

(1) 리빌딩 전송(DMA) 시간의 정의

통신시스템의 데이터 저장용 보드의 비휘발성 메모리를 하드디스크로 교체 시 가장 문제시 되는 것은 전송속도 문제이다. 보통 2GB를 전송시 10분 이내로 전송하여야 하며, 리빌딩 도중 호스트 서비스를 위해 한번에 몇 개의 섹터씩 나누어 전송한다. 리빌딩 타이밍도와 시간에 대한 정의를 그림 5와 표 3에서 각각 보여주고 있다.

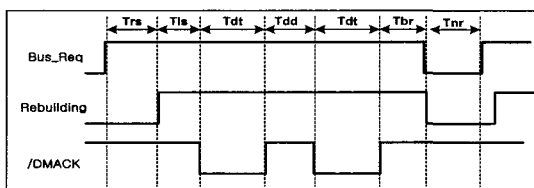


그림 5. 리빌딩 타이밍도

표 3. 빌딩 타이밍 신호의 정의

Define	Parameter Description	Time
Trs	Request Setup Time	24ns
Tis	IDE Register Setup Time	850μs
Tdt	DMA Transfer(1 sector)	46.3μs
Tdd	DMARQ Delay Time	2.8μs
Tbr	Bus Release Time	5.75ms
Tnr	Next Request Time	5.9μs

위의 신호를 이용하여 전송되는 시간의 식을 구하면 아래와 같다.

$$\frac{TTS}{DS} - \left[\frac{TTS}{DS} \right] \neq 0 \text{ 이면 } RN = \left[\frac{TTS}{DS} + 1 \right]$$

$$\frac{TTS}{DS} - \left[\frac{TTS}{DS} \right] = 0 \text{ 이면 } RN = \frac{TTS}{DS}$$

$$Ttr = RN \{ Trs + Tis + Tbr + Tnr \} + (RN - 1) Tnr$$

$$+ DS \cdot Tdt + (DS - 1) Tdd$$

$$= RN \{ (Trs + Tis + Tbr + Tnr) + DS(Tdt + Tdd) \}$$

$$\approx RN(6.6ms + DS \cdot 49.1\mu s)$$

- DS(Default Sector) : 한번에 전송할 섹터 수
- TTS(Total Transfer Sector) : 전체 섹터 수
- RN(Rebuilding Number) : 총 리빌딩 횟수
- Ttr(Total Transfer Sector) : 총 전송 시간

(2) 하드디스크 용량에 따른 전송 시간

표 4는 한번에 전송하는 섹터 수를 100으로 했을 때 하드디스크의 용량에 따라 소요되는 전송시간은 1GB시 4.6분, 2GB시 7.6분으로 10분이내의 전송이 가능함을 나타낸다.

표 4. 용량증가에 따른 전송시간의 변화(단위 전송 섹터 : 100섹터)

Capacity	Transfer Rate	Transfer Time	Transfer Time (min)
1G	20×10 ⁶	24×10 ³	460
1.2G	24×10 ⁶	24×10 ³	460
1.4G	28×10 ⁶	28×10 ³	537
1.6G	32×10 ⁶	32×10 ³	613
1.8G	36×10 ⁶	36×10 ³	690
2G	40×10 ⁶	40×10 ³	767
2.5G	50×10 ⁶	50×10 ³	959
3G	60×10 ⁶	60×10 ³	1151

(3) 한번에 전송할 섹터 수에 따른 전송시간

그림 6은 하드디스크의 용량을 2GB로 정의했을 때 단위 전송섹터의 수에 따라 소요되는 전송시간은 50섹터, 100섹터, 200섹터에 따라 각각 12분, 7.6분, 5.4분으로 시간이 감소함을 나타낸다.

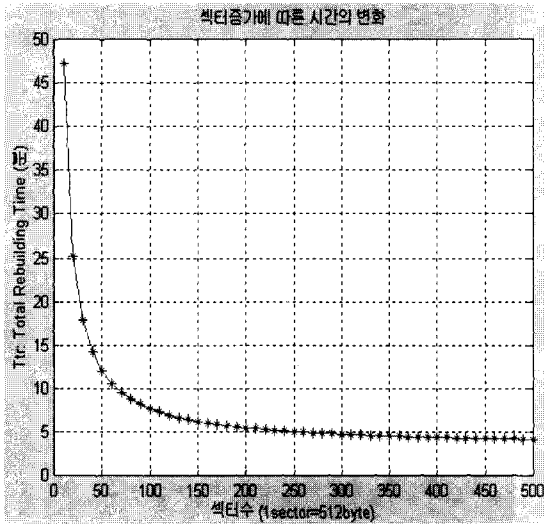


그림 6. 섹터의 증가에 따른 전송시간의 변화

V. 결론

교환시스템에서 사용할 수 있는 기억장치로는 플래시 메모리나 배터리백업 메모리를 사용할 수 있으나 단위 용량당 가격이 높아 증가하는 데이터를 모두 저장하기 위해서는 많은 비용을 필요로 한다. 따라서 본 연구에서는 플래시메모리보다 가격이 저렴한 소형 하드디스크를 이용하여 저장장치를 구현하였으며, 안정적이고 신뢰성 있는 서비스를 제공하기 위해 이중화를 지원하도록 설계하였다.

본 연구에서는 구현된 이중화 저장장치는 연속적으로 신뢰성 있는 서비스를 제공하기위해서 시스템의 동작 중에 디스크를 교체했을 때 기존의 데이터를 빠른 속도로 백업시켜주는 리빌딩 기능, 디스크에서 베드섹터가 발생하였을 때 자동적인 섹터 리맵핑기능, 리빌딩시 호스트 서비스 기능, 서비스의 우선순위 지정 기능 등을 제공하는 이중화 저장장치를 개발하였다. 또한 리빌딩 도중 발생한 에러를 복구하기위해 리맵핑과정을 수행함으로써 데이터의 신뢰성을 향상시키도록 하였다. 또한 하드디스크로 교체시 발생하는 속도문제는 2GB기준으로 100섹터

씩 전송할 때 10분 이내에 전송됨을 구현결과를 통해 확인하였다. 본 연구를 통해 얻게 될 윈 보드 타입의 소형 디스크 모듈은 교환기 시스템의 여러 보드의 백업장치를 저 가격에 다양한 디스크 컨트롤 기능을 제공하여 시스템의 보다 효율적이고 안정적인 서비스를 제공하는데 기여 할 것으로 기대된다.

참고 문헌

- [1] Paul Massiglia, *The RAID book 6th edition : A storage System Technology Handbook*, RAID Advisory Board, 1997.
- [2] Design and implementation of disk array controller, Proceedings of KITE Fall Conference '96 1996, 11 v.19, n.2, pt.B, pp.665-668.
- [3] Intel StrataFlash Memory(J3) Datasheet, Intel, Jan. 2004.
- [4] CompactFlash Card Specification, Transcend Information Inc.
- [5] Hard disk drive specifications, Hitachi Microdrive, Hitachi Global Storage Technologies, Jan. 2003.
- [6] Hitachi DK23EB-40 Disk Drive Specification, Hitachi Ltd, Feb. 2002.
- [7] Friedhelm Schmidt " The SCSI Bus and IDE interface Protocols, applications and programming " Addison-Wesley 1998.
- [8] Working Draft proposed american national standard "information technology-AT Attachment with PacketInterface Extension (ATA/ATAPI-4) document" Revision17 1997.
- [9] Intel, "Embedded Microcontrollers and Processors" Intel Corporation, 1995.
- [10] I. Scott Mackenzie "The 8051 Microcontroller" Prentice Hall.

노 승 환(Soonghwan Ro)

정회원



1987년 : 고려대학교 공학사
1989년 : 고려대학교 공학석사
1993년 : 고려대학교 공학박사
1997년 ~ 1998년 : 한국전자
통신연구원 초빙연구원
2002년 ~ 2003년 : 영국
Birmingham대학교

전기전자 & 컴퓨터 공학과 연구원

1994년~현재 : 공주대학교 정보통신공학부 부교수
<주관심분야> Pervasive computing, 임베디드 시스템, 모바일 IP