

# AFA(All-Flash Array) 탑재 서버의 에너지 효율성에 대한 연구\*

김영만\*\* · 한재일\*\*\*

## A Study on Energy Efficiency in Servers Adopting AFA(All-Flash Array)\*

Young Man Kim\*\* · Jaeil Han\*\*\*

### ■ Abstract ■

Maximizing energy efficiency minimizes the energy consumption of computation, storage and communications required for IT services, resulting in economic and environmental benefits. Recent advancement of flash and next generation non-volatile memory technology and price decrease of those memories have led to the rise of so-called AFA (All-Flash Array) storage devices made of flash or next generation non-volatile memory. Currently, the AFA devices are rapidly replacing traditional storages in the high-performance servers due to their fast input/output characteristics. However, it is not well known how effective the energy efficiency of the AFA devices in the real world. This paper shows input/output performance and power consumption of the AFA devices measured on the Linux XFS file system via experiments and discusses energy efficiency of the AFA devices in the real world.

Keyword : All-Flash Array, NVMe, NVDIMM, Energy Efficiency, File System

Submitted : November 9, 2018

1<sup>st</sup> Revision : December 20, 2018

Accepted : February 15, 2019

\* 본 논문은 정보통신기술진흥센터의 지원으로 수행되었음(No. R0101-15-237, 고집적 저전력 프로세서 기반 30% 이상 에너지 절감 범용 운영체제 및 가상화 핵심기술 개발).

\*\* 국민대학교 소프트웨어학부 교수

\*\*\* 교신저자, 국민대학교 소프트웨어학부 교수

## 1. 서론

최근 IoT(Internet of Things), 클라우드 컴퓨팅, 빅데이터 처리 등의 기술 발전에 따라 이를 활용한 IT서비스가 확산하고 있으며, 사용자 증가에 따라 이를 지원하기 위한 컴퓨팅 및 통신 자원이 빠르게 증가하여 전력 소모도 크게 증가하고 있다(Agrawal, 2011; Armbrust, 2010). 따라서 이러한 전력 소모를 줄일 수 있는 에너지 효율적 시스템은 그런 IT서비스에 매우 중요하며 중요한 이슈가 되고 있다(Set, 2017).

에너지 효율성(Energy Efficiency)은 작업/에너지(또는 성능/과워)를 의미하며, 동일 작업에 더 적은 에너지의 사용을 목표로 한다. 컴퓨팅 시스템은 컴퓨터 사용률이 가장 높을 때 최고의 에너지 효율성을 달성하나, 사용률 감소에 비례하여 전력 소모가 감소하지 않고 사용률보다 적게 감소하여 에너지 효율성이 급격히 나빠지게 된다. 따라서 IT서비스에 소모되는 에너지를 최소화하여 경제적이고 환경적인 이익을 가져오기 위해, 부하가 감소할 경우에도 수행 작업에 비례하여 최적의 에너지 효율성을 유지할 수 있게 해주는 에너지 비례 시스템(Energy-Proportional System)이 중요한 기술로 대두되고 있다(Set, 2017; Wu, 2016; Acun, 2016).

최근 플래시와 차세대 비휘발성 메모리의 기술 발전 및 가격 저하로 인해 플래시나 차세대 비휘발성 메모리로 만든 소위 AFA(All-Flash Array) 저장장치가 부상하고 있다(Tech1; Tech2). AFA는 반도체 칩으로 만들어지며, 기존의 SSD가 하드 디스크 인터페이스를 통해 접근해야 하는 것과 달리 메모리에 직접 접근(access)하여 빠른 데이터 입·출력을 할 수 있어 고성능 서버 중심으로 로컬 스토리지를 AFA로 대체하는 움직임이 빨라지고 있다(Tech2). 이러한 환경에서의 에너지 비례 시스템은 AFA에 최적화된 에너지 절감 관리 수단과 방법을 갖춘 운영체제가 필요하다(Set, 2017; Wu, 2016).

본 논문은 AFA에 최적화된 에너지 절감 관리

정책 개발에 필요한 기초 연구로서 기존 메모리인 HDD(Hard Disk Drive)와 SATA(Serial ATA) 방식 SSD(Solid-State Drive), 그리고 최신 비휘발성 메모리 중 NVMe(Non-Volatile Memory Express)와 NVDIMM(Non-Volatile Dual In-line Memory Module) 메모리를 사용하여 서로 다른 파일 크기, CPU 성능, 시스템 함수의 호출 수, 버퍼 크기 등 여러 변수를 대상으로 쓰기(Write)/읽기(Read) 실험을 통해 입·출력 성능과 소비 전력량을 측정하고 결과를 비교 분석한다.

본 논문의 구성은 다음과 같다. 제 2장은 배경 및 관련 연구를 기술하고, 제 3장은 실험 환경을 논한다. 제 4장은 실험 결과에 대해 설명하고, 제 5장은 결론을 기술한다.

## 2. 배경 및 관련 연구

비휘발성 메모리는 전원이 지속적으로 공급되지 않아도 정보를 기억할 수 있는 메모리이다. 기존의 비휘발성 메모리 저장장치 중 대표적인 것은 HDD와 플래시 메모리 기반의 SATA 방식 SSD이며 입출력 속도가 느리다. 휘발성 메모리는 입출력 속도는 빠르나 전원이 공급되지 않으면 정보가 분실된다. 차세대 비휘발성 메모리는 전원이 공급되지 않아도 정보를 저장할 수 있으며 속도도 휘발성 메모리에 근접하여 기존 비휘발성 메모리와 휘발성 메모리의 장점을 제공한다. 또한 소비 전력이 적기 때문에 에너지 절감 효율을 높일 수 있다. 이러한 차세대 비휘발성 메모리 중 최근 출시된 저장형태로는 NVMe, NVDIMM 등이 있다(Zhang, 2015; Sueti, 2014).

### 2.1 HDD와 SSD

HDD는 비휘발성, 순차접근이 가능한 컴퓨터의 보조 기억 장치이다. 보호 케이스 안에 있는 디스크 크기가 빠른 속도로 회전하며, 자기 정보를 기록하는 헤드가 이동하여 디스크에 자기 패턴으로 정보를

기록한다. 디스크와 헤드의 물리적인 이동이 필요하기 때문에 속도가 느리나 저장용량 당 가격이 저렴하여 여전히 중요한 저장매체로 사용되고 있다.

SSD는 반도체를 이용하여 정보를 저장하는 저장장치이다. 순수 전자식으로 작동하므로 기계적인 HDD의 문제인 긴 탐색 시간, 반응 시간, 기계적 지연, 실패율, 소음을 크게 줄여 주며 HDD보다 빠른 읽기·쓰기 속도를 제공한다.

## 2.2 NVMe 메모리

기존의 SSD는 AHCI(Advanced Host Controller Interface)와 SATA 방식을 이용하며, 다양한 소프트웨어와 호환된다는 장점이 있다. 그러나 AHCI 특성상 SSD와 호환성이 떨어지며 병목 현상이 발생할 수 있다. NVMe 메모리는 PCIe(Peripheral Component Interconnect Express) 방식으로 연결된 비휘발성 저장장치로써 위 문제를 개선하기 위하여 AHCI보다 높은 대역폭을 제공한다(Zhang, 2015).

## 2.3 NVDIMM 메모리

NVDIMM(Non-Volatile Dual-Inline Memory Module)은 비휘발성 메모리를 탑재한 DIMM 규격의 저장장치로써 비휘발성 RAM(Random Access Memory)을 구현한다(Suei, 2014). NVDIMM 메모리 모듈은 휘발성 메모리인 DRAM(Dynamic RAM)을 탑재할 수 있으며, 이 경우 DRAM은 버퍼 역할을, NAND 플래시 메모리는 백업 저장장치의 역할을 하여 만약 NVDIMM의 전원이 차단될 경우, DRAM에 있는 데이터를 플래시 메모리에 자동으로 전송하는 기능과, 전원이 돌아왔을 때 플래시 메모리의 데이터를 DRAM에 되돌리는 기능이 제공된다(Suei, 2014; Netlist1).

NVDIMM은 하드웨어 구성 및 작동 특성에 따라 크게 NVDIMM-N, NVDIMM-F, NVDIMM-P 등의 세 가지 유형으로 분류된다(Jedec). NVDIMM-

N은 DRAM과 플래시 메모리가 같은 비율로 탑재되며, 플래시 메모리는 메모리 모듈 내부에서 백업 메모리의 역할을 한다. NVDIMM-F는 DRAM 없이 비휘발성 메모리만 탑재한 메모리 모듈이다. CPU나 OS는 이 메모리 모듈을 스토리지로 취급하며 스토리지 내부의 비휘발성 메모리를 직접 액세스할 수 있다. 또한 DRAM DIMM 규격의 메모리에서 불가능한 큰 저장 용량을 제공한다. 그러나 액세스 지연시간(latency)이 NVDIMM-N에 비하여 매우 길다. NVDIMM-P는 RRAM(Resistive RAM)과 같은 차세대 비휘발성 메모리를 탑재한다. 이러한 차세대 비휘발성 메모리는 DRAM과 같은 속도로 임의의 메모리 장소를 접근할 수 있어 비휘발성 DRAM으로 간주되며, 컴퓨터 주기억장치에서의 스토리지 사용을 가능하게 한다. 그러나 NVDIMM-P 메모리는 현재 표준화 작업이 완료되지 않았으며, 상용화된 NVDIMM 메모리 모듈은 대부분 플래시 메모리를 사용하고 있다. 본 연구에서는 NVDIMM-N 유형의 메모리가 기능적으로 DRAM과 거의 같아 실험에 NVDIMM-N 유형의 메모리를 사용하였다.

## 2.4 기존 연구 및 문제점

기존 또는 차세대 비휘발성 메모리를 스토리지나 주기억장치 등에 사용하여 시스템 성능을 개선하기 위한 많은 연구가 수행되었다(Park, 2008; Condit, 2009; Caulfield, 2012; Choi, 2013). 그러나 이 연구들은 이론적인 관점에서 비휘발성 메모리 활용이 파일 시스템의 성능에 미치는 영향 분석에 초점을 두고 있으며, 서로 다른 비휘발성 메모리의 상용 제품을 탑재한 실제 운용 환경에서 얻어지는 시스템 성능이나 에너지 절감에 대한 연구는 Han(2017)에서 일부 언급되었을 뿐 매우 부족한 실정이다. 다양한 비휘발성 메모리에 대한 실용성 분석은 에너지 비례 시스템의 효과적인 에너지 관리 정책 수립에 큰 영향을 미치는 요소로서 심도 있는 연구가 필요하다.

### 3. 실험 환경

본장에서는 실험에 사용된 서버의 하드웨어 구성 및 운영체제를 설명하고, 비휘발성 저장장치의 성능 및 에너지 효율성을 측정하기 위하여 분석한 리눅스 커널의 파일시스템 구조와 시스템 함수 호출에 대하여 기술한다.

#### 3.1 서버 하드웨어 구성 및 운영체제

실험은 기존의 대표적인 비휘발성 저장장치인 HDD와 SATA 방식 SSD, 그리고 차세대 비휘발성 메모리 중 최근 출시된 PCIe 방식 NVMe SSD와 DDR4 방식 NVDIMM-N 등의 네 종류 저장장치를 대상으로 하였으며, 실험에 사용한 저장장치의 상세 내역은 다음과 같다.

- HDD : WD10EZEX(1TB) (Wdc)
- SATA 방식 SSD : Samsung SSD 750 EVO (120GB) (Samsung)
- NVMe SSD : Intel Solid State Drive 750 Series (400GB) (Intelssd)
- N유형 NVDIMM : NETLIST NVvault DDR4 (NV4) (8GB) (Netlist2; Netlist3)

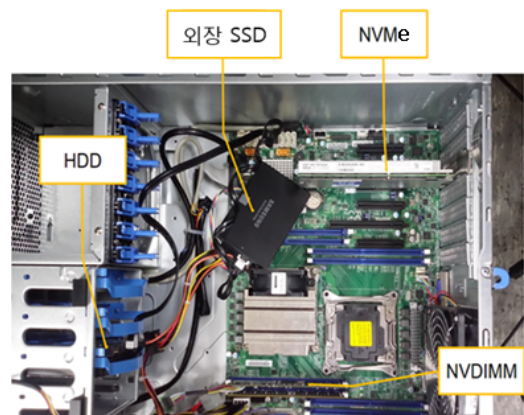
실험은 CPU 성능이 서로 다른 비휘발성 저장장치로 작동하는 시스템의 성능에 미치는 영향을 알기 위해 클럭 수가 1.6GHz와 2.4GHz인 두 종류의 CPU를 사용하였다. 주기관은 NVMe와 NVDIMM 저장장치를 지원하는 Intel C612 칩셋(chipset)을 탑재한 특수 주기관을 사용하였고, 주기억장치는 주기관 사양인 DDR4 규격의 DRAM을 사용하였다. 실험에 사용된 CPU, 주기억장치, 주기관 등 서버를 구성하는 컴포넌트의 상세 내역은 다음과 같다.

- CPU1 : Intel Xeon Processor E5-2620 v3 (2.4GHz, 15M Cache) (Intel24)
- CPU2 : Intel Xeon Processor E5-2603 v3

(1.6GHz, 15M Cache) (Intel16)

- 주기관 : SUPERMICRO X10DRL-I
- 주메모리 : REG, ECC DDR4 SDRAM(2133MHz, 8GB)

실험에 사용한 운영체제는 리눅스 계열의 CentOS 7.2.1511(Kernel version 4.4.0)이다. <Figure 1>은 위 컴포넌트로 조립한 서버의 내부 모습이다.

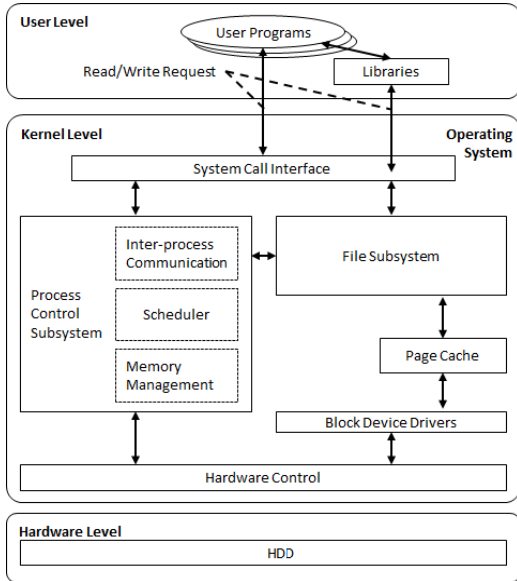


<Figure 1> Inside of a Server Computer

#### 3.2 성능 측정을 위한 변수

본 연구는 실험에 사용할 리눅스 XFS 파일시스템의 구조와 내부 시스템 호출을 분석하여 서로 다른 저장장치를 탑재한 시스템의 성능에 영향을 미칠 수 있는 두 개의 변수를 추출하였다. 하나는 리눅스 커널이 관리하는 페이지 캐시의 버퍼 크기이며(<Figure 2> 참조), 다른 변수는 출력할 때 버퍼를 실제 저장장치에 저장하는 작업량으로 fsync 명령 호출 회수에 비례한다. 작업량 변수는 입력과 관계없이 입력 성능 실험에서 제외하며 대신 서로 다른 자료형의 데이터, 즉 다른 크기의 데이터를 입력하는 실험을 수행한다.

실험은 위 변수에 더해 파일 크기, CPU 성능 등을 포함한 네 가지 변수를 대상으로, 각 저장장치를 사용했을 때 페이지 캐시로부터 저장장치로의 데이터 입력 시간 및 저장장치로부터 페이지



<Figure 2> CentOS 7 Linux Operating System

캐시로의 출력 시간을 측정한다. 본 실험에서는 저장장치의 입력 및 출력 시간을 보다 정확히 측정하기 위해 커널의 시스템 함수를 수정하여 리빌드한 커널을 사용하였다.

E를 에너지, P를 전력, T를 시간이라 할 때 에너지 소비는 다음 식에 의해 계산된다.

$$E = P \times T$$

따라서 각 저장장치의 소모 에너지는 성능 측정에서 얻은 저장장치 사용 시간과 각 메모리의 소비 전력을 곱하여 얻은 소비 전력량으로 계산한다. <Table 1>은 각 저장 장치에서 읽거나 쓰기 작업을 하는데 필요한 평균 소비 전력을 보이고 있다(Wdc; Samsung; Intelssd, Netlist3).

<Table 1> Average Power Consumption

	Read	Write
HDD	6.8W	6.8W
SSD	2.1W	3.7W
NVMe	9W	12W
NVDIMM	2W	2W

## 4. 입·출력 성능 및 에너지 효율성

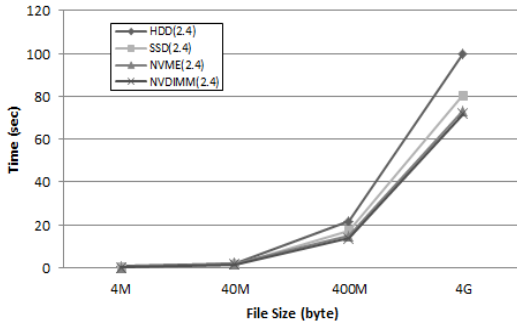
본 장은 HDD, SATA 방식 SSD, NVMe, NVDIMM 메모리로 구성된 각 저장장치를 사용했을 때 파일 크기, CPU 속도, 시스템 함수의 호출 수, 버퍼 크기, 데이터 자료형 등의 변수에 대해 CentOS 7 운영체제의 XFS 리눅스 파일시스템을 통한 정수형 데이터 입·출력으로 메모리의 성능을 측정하고 소비 전력량을 계산하여 에너지 효율성을 비교한다.

### 4.1 출력 성능 및 소모 에너지 비교

출력 실험은 파일 크기, CPU 속도, 시스템 함수의 호출 수, 버퍼 크기 등의 각 변수에 대해 저장할 데이터를 fwrite 또는 fprintf 명령어를 통해 출력하며, 출력 성능은 페이지 캐시에 있는 데이터를 저장장치로 출력하는 시간을 측정하여 평가한다.

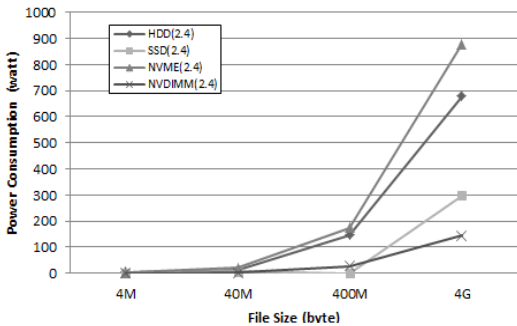
#### 4.1.1 파일 크기에 따른 실험 결과

<Figure 3>은 파일 크기를 4MB, 40MB, 400MB, 4GB로 증가시키면서 저장장치별로 쓰기 한 시간을 측정한 결과를 나타낸다(Han, 2017). 이 그림에서는 평균적으로 보았을 때 같은 파일 크기에 대하여 HDD의 쓰기 시간이 가장 오래 걸렸으며 이어서 SSD, NVMe, NVDIMM 순서로 시간이 줄어드는 결과를 보여준다. 4MB, 40MB와 같이 파일 크기가 작은 경우에는 저장장치별 속도 차이가 거의 없었으나, 400MB, 4GB와 같이 파일 크기가 커짐에 따라 저장장치별 속도 차이가 점점 벌어지고 있다. 예를 들어 HDD와 NVDIMM 사이의 속도 차이는 약 1.3배를 보여준다. 그러나 네 종류의 저장장치 성능을 고려할 때 기대보다 출력 속도의 차이가 적게 나타나고 있다. 이것은 기존 파일시스템이 HDD와 SSD와 같이 저속의 저장장치에 최적화되어 고속 저장장치인 NVMe와 NVDIMM 기반 저장장치의 출력 성능을 제대로 활용하지 못하는 것으로 분석된다.



<Figure 3> Write Time(File Size)

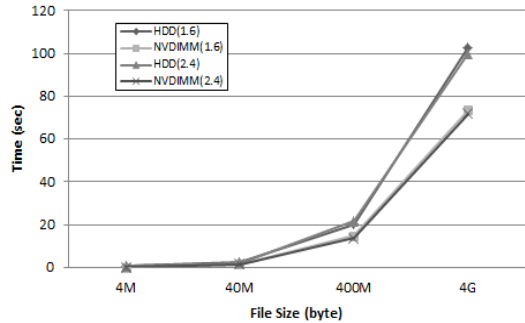
<Figure 4>는 파일 크기에 따른 쓰기 작업의 소비 전력량을 계산한 결과이다. 이 결과는 NVDIMM이 가장 적은 전력량을, NVMe가 가장 큰 전력량을 소비하고 있음을 보여주고 있다. NVMe의 쓰기 시간은 HDD보다 적으나 <Table 1>에서 보듯이 NVMe의 쓰기 작업은 HDD보다 약 2배에 이르는 전력이 필요하여 NVMe의 소비 전력이 가장 크게 나타난 반면, NVDIMM의 소비 전력은 NVMe의 약 1/7에 불과한 것으로 나타나고 있다.



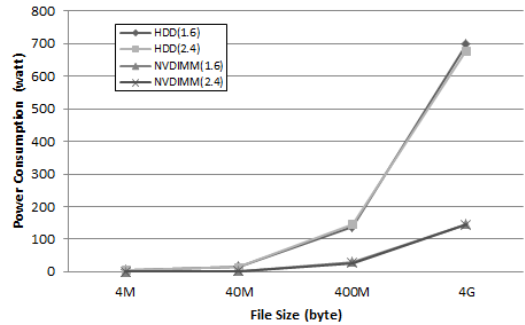
<Figure 4> Power Consumption(File Size)

#### 4.1.2 CPU 속도에 따른 실험 결과

<Figure 5>는 CPU 속도가 저장장치에 미치는 영향을 파악하기 위하여 제 4.1.1절과 같은 방법으로 실험하되 CPU 클럭 주기가 1.6GHz와 2.4GHz 인 두 개의 CPU를 사용하여 HDD와 NVDIMM 저장장치의 성능을 측정하였다. 제 4.1.1절에서 가장 큰 성능 차이를 보인 HDD와 NVDIMM의 실험 결과를 비교해 보면, 4GB 파일 크기에서 1.6GHz는



<Figure 5> Write Time(CPU Speed)



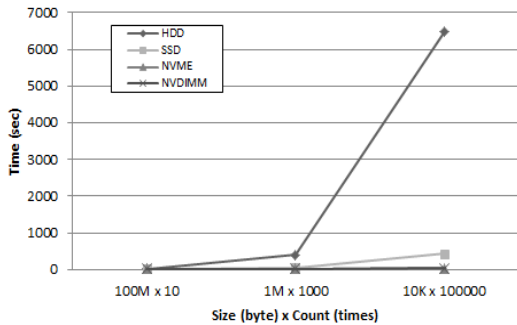
<Figure 6> Power Consumption(CPU Speed)

약 1.23배, 2.4GHz는 약 1.25배의 성능 차이를 보여 CPU 속도가 출력 성능에 미치는 영향은 거의 없음을 알 수 있다. 그래프에 나타내지 않았으나 SSD와 NVMe도 거의 차이가 없는 결과를 보여 주었다. 또한 소비 전력이 시간에 비례하므로 <Figure 6>에서 보듯이 CPU 속도가 소비 전력에 미치는 영향도 거의 없음을 알 수 있다.

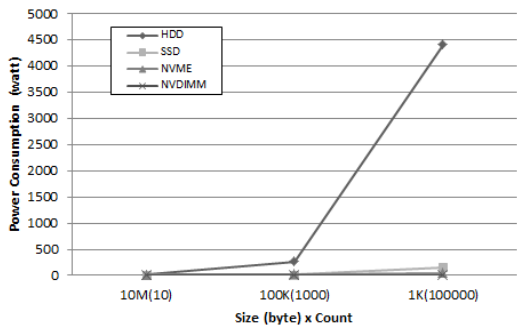
#### 4.1.3 fsync 함수 호출 수에 따른 실험 결과

버퍼를 실제 저장장치에 저장하는 작업량에 따른 실험은 저장할 데이터를 fwrite 또는 fprintf 명령어를 통해 페이지 캐시의 출력 버퍼(Write Buffer)에 쓰기(Write)한 뒤, 버퍼에 저장된 데이터를 fsync 명령어로 최종 저장장치에 저장하는 방법으로 수행되었다(Agrawal, 2008).

<Figure 7>은 1GB로 파일의 크기를 고정하고, fsync 함수 호출 수를 늘리면서 저장장치의 성능을 측정한 그래프이다. fsync 함수는 호출되는 순



<Figure 7> Write Time(Fsync Call Count)



<Figure 8> Power Consumption(Fsync Call Count)

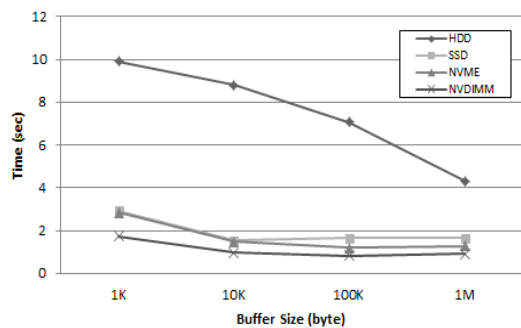
간 출력 버퍼에 있는 데이터를 바로 저장장치에 저장(Write)하여 데이터를 동기화 시킨다. 실험 결과를 보면 HDD는 fsync 호출 수가 늘어날수록 시간이 크게 증가하여, fsync 호출 수가 1,000번 일 때와 100,000번 일 때의 차이는 약 13배이다. SSD와 NVMe는 fsync 호출 수가 커질수록 시간이 적은 폭으로 증가한다. NVDIMM에서는 fsync 호출 수를 늘려도 시간의 차이가 거의 없었다. 실험 결과는 NVDIMM이 sync 호출 수에 상관없이 다른 저장장치에 비해 항상 성능이 우수함을 보이고 있다.

<Figure 8>은 파일 크기를 1GB로 고정하고 fsync 호출 수에 따른 소비 전력량을 측정된 그래프이다. SSD와 NVMe의 소비 전력량 차이는 거의 없으나, HDD와 NVDIMM의 소비 전력량 차이는 최대 800배 정도의 차이를 보였다. HDD의 소비 전력량이 큰 폭으로 증가한 이유는 fsync 호출 수가 증가 할수록 실제 저장장치에 쓰기 작업이 많이 수행되는데, 기계적인 쓰기 작업을 하는 HDD

에서는 병목현상이 일어나 그 속도가 현저히 느려져 쓰기 작업 시간이 많이 증가하므로 이에 비례하여 소비 전력량의 크기가 매우 커지기 때문이다. fsync 호출 수가 1,000번일 때 HDD와 SSD의 차이는 8.9배, HDD와 NVMe는 25배, NVDIMM과는 26배 정도의 소비 전력 차이를 보인다.

#### 4.1.4 출력 버퍼 크기에 따른 실험 결과

<Figure 9>는 1GB로 파일 크기를 고정하고 출력 버퍼의 크기를 증가 시키면서 저장장치의 성능을 측정된 그래프이다. HDD는 출력 버퍼의 크기를 증가 시킬수록 저장장치에 버퍼 데이터 쓰기 작업이 줄어들어 시간이 감소한다. 다른 저장장치들은 버퍼 크기가 10KB로 커질 때까지는 시간이 감소하나 10KB보다 커질 경우 거의 차이가 없다. 출력 버퍼의 크기가 1KB일 때 HDD는 SSD와 NVMe보다 약 4배, NVDIMM보다 약 5.2배의 시간이 소모된다. 출력 버퍼 크기가 1MB일 때는 HDD가 SSD보다 2.5배, NVMe보다 2.6배, NVDIMM보다 2.4배의 시간이 소모된다. 소비 전력은 NVMe 쓰기 작업의 소모 전력이 많아 NVDIMM, SSD, NVMe, HDD 순서로 증가한다.



<Figure 9> Write Time(Write Buffer Size)

## 4.2 입력 성능 및 소모 에너지 비교

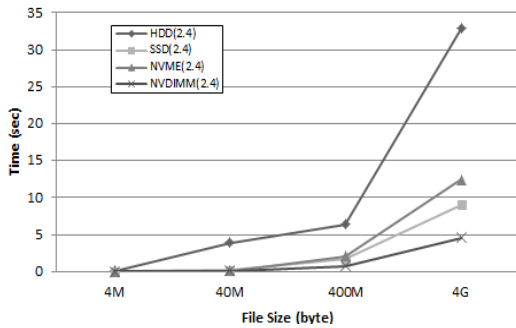
입력 실험은 파일 크기, CPU 속도, 버퍼 크기, 입력 자료형 등의 각 변수에 대해 데이터를 fread 명령어를 통해 입력하며, 입력 성능은 저장장치로부터 페이지 캐시로 데이터를 입력하는 시간을 측정

하여 평가한다. fread로 데이터를 입력할 때 데이터가 계속 버퍼에 남아있는 것을 읽기(Read)해서 성능 측정이 제대로 되지 않는 오류가 발생할 수 있으므로 캐시를 비우는 명령을 실행하고 입력 시간을 측정하였다.

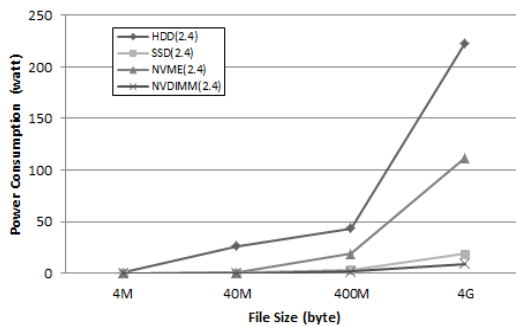
4.2.1 파일 크기에 따른 실험 결과

<Figure 10>은 파일 크기를 4MB, 40MB, 400MB, 4GB로 증가시키면서 저장장치 별로 입력한 시간을 측정한 결과이다. 평균적으로 보았을 때 같은 파일 크기에 대하여 HDD의 입력 시간이 가장 길며 이어서 NVMe, SSD, NVDIMM 순서로 시간이 줄어든다. 4GB 파일을 읽었을 때 입력 속도가 NVDIMM은 HDD보다 약 6.9배 빠르나 NVMe는 SSD보다 더 느리게 나타난다. 이것은 CentOS의 파일 시스템이 기존의 저속 저장장치에 최적화되어 고속 저장장치의 입력 성능을 제대로 지원하지 못하기 때문으로 분석된다.

<Figure 11>은 파일 크기에 따른 입력 작업의



<Figure 10> Read Time(File Size)

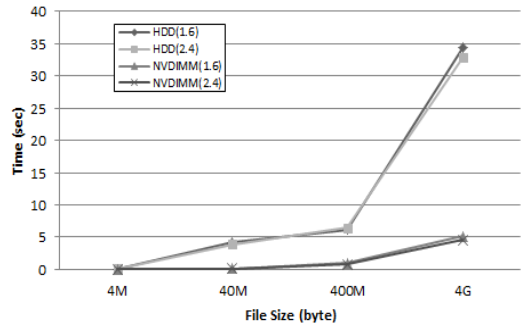


<Figure 11> Power Consumption(File Size)

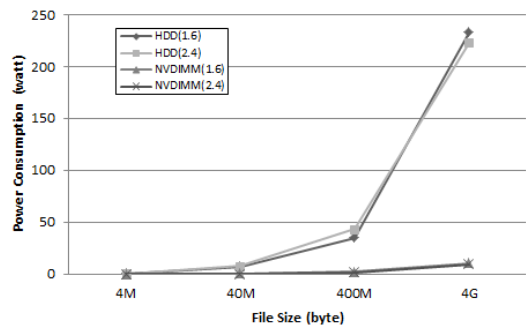
소비 전력량을 나타낸다. 소비 전력은 HDD, NVMe, SSD, NVDIMM 순서로 감소되며, 4GB 파일을 읽을 때 NVDIMM의 소비 전력이 HDD보다 약 7.18배 적은 것으로 나타났다.

4.2.2 CPU 속도에 따른 실험 결과

<Figure 12>는 CPU 속도가 저장장치에 미치는 영향을 파악하기 위하여 제 4.2.1결과 같은 방법으로 실험하되 CPU 클럭 주기가 1.6GHz와 2.4GHz인 두 개의 CPU를 사용하여 HDD와 NVDIMM 저장장치의 성능을 측정하였다. 제 4.2.1절에서 가장 큰 성능 차이를 보인 HDD와 NVDIMM의 실험 결과를 보면, CPU 속도가 1.6GHz에서 2.4GHz로 1.5배 증가했음에도 불구하고 읽기 시간은 거의 차이가 없다. 그래프에 나타내지 않았으나 SSD와 NVMe도 거의 차이가 없는 결과를 보였다. 또한 소비 전력은 시간에 비례하므로 <Figure 13>에서 보듯이 CPU 속도가 소비 전력에 미치는 영향도 거의 없음을 알 수 있다.

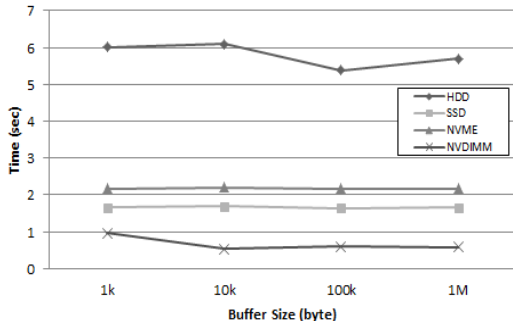


<Figure 12> Read Time(CPU Speed)

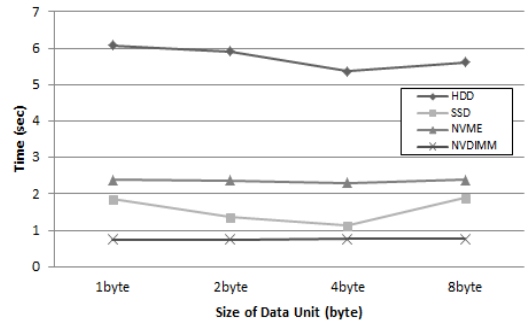


<Figure 13> Power Consumption(CPU Speed)

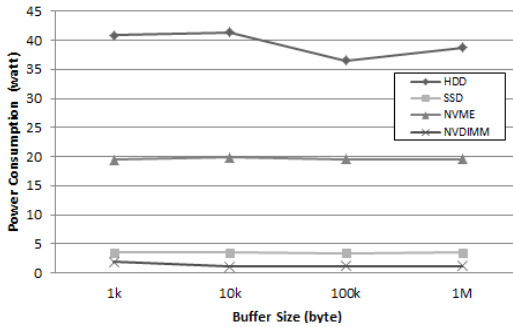




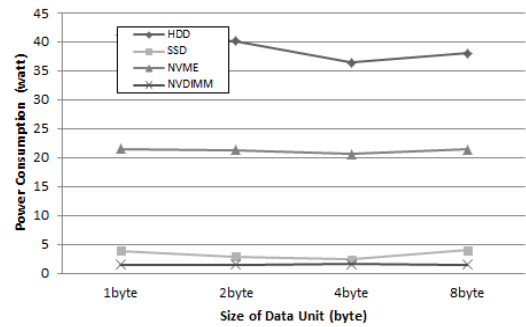
<Figure 14> Read Time(Read Buffer Size)



<Figure 16> Read Time(Data Type)



<Figure 15> Power Consumption(Read Buffer Size)



<Figure 17> Power Consumption(Data Type)

#### 4.2.3 입력 버퍼 크기에 따른 실험 결과

<Figure 14>는 1GB로 파일 크기를 고정하고 입력 버퍼의 크기를 증가 시키면서 저장장치의 성능을 측정하는 것이다. 각 저장장치의 입력 성능 차이로 서로 다른 저장장치의 입력 시간은 차이가 있으나 저장장치별 입력 성능은 입력 버퍼 크기에 따라 큰 차이가 없는 것으로 나타났다. <Figure 15>는 입력 버퍼 크기에 따른 소비 전력을 나타내며, HDD는 소비 전력이 다소의 편차가 있으나 다른 저장장치들은 거의 차이가 없는 결과를 보이고 있다.

#### 4.2.4 입력 자료형에 따른 실험 결과

<Figure 16>은 파일 크기를 1GB로 고정하고, 1byte(char 형), 2byte(short 형), 4byte(int 형), 8byte(double 형)의 크기가 서로 다른 네 개의 자료형을 대상으로 실험한 결과를 나타낸다. NVMe와 NVDIMM은 자료형에 상관없이 거의 같은 성능을 보였다. HDD는 가장 느린 경우인 1byte와

가장 빠른 경우인 4byte가 약 1.13배의 차이를 보이며, SSD는 가장 느린 경우인 8byte와 가장 빠른 경우인 4byte가 약 1.67배의 차이를 보였다. <Figure 17>은 자료형에 따른 소비 전력을 나타내며 입력 성능 결과와 같이 소비 전력이 적게 드는 순서도 동일하였다.

## 5. 결 론

가속화하고 있는 IT 기술 발전은 새로운 IT 서비스의 출현과 확산을 촉진시키고 있으며, 사용자 증가에 따라 컴퓨터, 통신, 스토리지 등의 더 많은 컴퓨팅 자원이 요구되어 에너지 소모도 증가하게 된다. 이에 따라 모바일 디바이스, PC, 데이터 센터의 서버 등 다양한 IT 기기에서 최고의 에너지 효율성을 유지할 수 있는 기술 개발이 요구된다.

최근 플래시 및 차세대 비휘발성 메모리 기반의 저장장치 용량 증대와 가격 저하로 서버 환경에서

로컬 스토리지를 이러한 비휘발성 메모리 기반의 AFA로 대체하는 움직임이 빨라지고 있다. 본 논문은 이러한 환경에서의 에너지 효율적 시스템 개발을 위한 기초 연구로 최신 상용 비휘발성 메모리를 사용한 실제 실험을 통해 입출력 성능과 소비 전력량을 측정하여 에너지 효율성을 분석하였다. 실험 결과는 NVMe 메모리나 NVDIMM과 같은 차세대 비휘발성 메모리 기반의 새로운 AFA 저장장치가 기존 저장장치(HDD, SSD)보다 에너지를 최대 7배까지 절감시키고, 또한 성능도 최대 8~9배까지 향상시키는 것을 보여 주었다.

그러나 본 논문은 저속 저장장치에 최적화된 기존 파일 시스템을 사용하여 NVMe와 NVDIMM 메모리 기반의 고속 저장장치에 대한 성능과 에너지 소모에 대한 결과가 제한적이며 기대와 다른 경우도 나타났다. 앞으로 차세대 비휘발성 메모리의 특성을 충분히 활용할 수 있도록 지원하는 여러 운영체제를 대상으로 실제 운용 환경에서의 실험을 통해 AFA 저장장치를 사용한 시스템의 성능과 에너지 효율성에 대한 평가가 객관적으로 이루어질 수 있도록 많은 연구가 필요하다.

## References

- Acun, B. et al., “Power, Reliability, and Performance : One System to Rule Them All”, *Computer*, Vol.49, No.10, 2016, 30-31.
- Agrawal, D., S. Das, and A. El Abbadi, “Big data and cloud computing : current state and future opportunities”, Proceedings of the 14<sup>th</sup> International Conference on Extending Database Technology, Uppsala, ACM, 2011, 530-533.
- Agrawal, N. et al., “Design tradeoffs for SSD performance”, USENIX 2008 Annual Technical Conference, Boston, Massachusetts, June 2008, 57-70.
- Armbrust, M. et al., “A view of cloud computing”, *Communications of the ACM*, Vol. 53, No.4, 2010, 50-58.
- Caulfield, A.M. et al., “Providing safe, user space access to fast, solid state disks”, Proceedings of the seventeenth international conference on Architectural Support for Programming Languages and Operating Systems, London, ACM, 2012, 387-400.
- Choi, G.S., B. On, K. Choi, and S. Yi, “PTL : PRAM Translation Layer”, *Microprocessors and Microsystems*, Vol.37, 2013, 24-32.
- Condit, J. et al., “Better I/O through byte-addressable, persistent memory”, in Proceedings of the ACM SIGOPS 22<sup>nd</sup> symposium on Operating systems principles, Big Sky, ACM, 2009, 133-146.
- Han, J. and Y. Kim, “A Study on the Performance and Energy Efficiency in the AFA-based Computing Server Environment”, KITS 2017 Fall Conference, Seoul, Korea, 2017, 419-422.
- IntelI6, [https://ark.intel.com/ko/products/83349/Intel-Xeon-Processor-E5-2603-v3-15M-Cache-1\\_60-GHz?q=xeon%20e5-2603%20v3](https://ark.intel.com/ko/products/83349/Intel-Xeon-Processor-E5-2603-v3-15M-Cache-1_60-GHz?q=xeon%20e5-2603%20v3).
- Intel24, [https://ark.intel.com/ko/products/83352/Intel-Xeon-Processor-E5-2620-v3-15M-Cache-2\\_40-GHz](https://ark.intel.com/ko/products/83352/Intel-Xeon-Processor-E5-2620-v3-15M-Cache-2_40-GHz).
- Intelssd, <http://www.intel.com/content/dam/www/public/us/en/documents/product-specifications/ssd-750-spec.pdf>.
- Jedec, <https://www.jedec.org/>.
- Netlist1, Netlist NVvault DDR4 NVDIMM with pmem Driver V0.3.pdf.
- Netlist2, <http://www.netlist.com/products/vault-memory-storage/nvvault-ddr4-nvdimm/default.aspx>.
- Netlist3, Netlist DDR4 NVvault Operational

- Guide V1.0p.pdf.
- Park, Y. et al., "PFFS : a scalable flash memory file system for the hybrid architecture of phase-change RAM and NAND flash", Proceedings of the 2008 ACM symposium on Applied computing, Fortaleza, ACM, 2008, 1498-1503.
- Samsung, [http://www.samsung.com/semiconductor/minisite/ssd/downloads/document/Samsung\\_SSD\\_750\\_EVO\\_Data\\_Sheet\\_Rev\\_2\\_1.pdf](http://www.samsung.com/semiconductor/minisite/ssd/downloads/document/Samsung_SSD_750_EVO_Data_Sheet_Rev_2_1.pdf).
- Set, R. and D.A. Wood, "Energy- Proportional Computing: A New Definition", *Computer*, Vol.50, No.8, 2017, 26-32.
- Suei, P., M. Yeh, and T. Kuo, "Endurance-Aware Flash-Cache Management for Storage Servers", *IEEE Transactions on Computers*, Vol.63, No.10, 2014, 2416-2430.
- Tech1, <http://searchsolidstatestorage.techtarget.com/definition/Flash-array>.
- Tech2, <https://www.techrepublic.com/article/all-flash-arrays-the-smart-persons-guide/>.
- Wdc, [https://www.wdc.com/content/dam/wdc/website/downloadable\\_assets/eng/spec\\_data\\_sheet/2879-771436.pdf](https://www.wdc.com/content/dam/wdc/website/downloadable_assets/eng/spec_data_sheet/2879-771436.pdf).
- Wu, X. et al., "Using Performance-Power Modeling to Improve Energy Efficiency of HPC Applications", *Computer*, Vol.49, No.10, 2016, 20-21.
- Zhang, Y. and S. Swanson, "A study of application performance with non-volatile main memory", 31<sup>st</sup> Symposium on Mass Storage Systems and Technologies(MSST), Santa Clara, CA, USA, 2015, 1-10.

## ◆ About the Authors ◆



**Young Man Kim (ymkim@kookmin.ac.kr)**

Young Man Kim received a BSc in Mechanical Engineering from Seoul National University, and an MSc and a PhD from Ohio State University. He is currently a professor at School of Computer Science, Kookmin University. His current research focuses on computer network and middleware. His research interests include computer network security, web frontend & backend framework, and IoT.



**Jaeil Han (jhan@kookmin.ac.kr)**

Jaeil Han received a BSc in Mathematics from Yonsei University, and an MSc and a PhD from Syracuse University. He is currently a professor at School of Computer Science, Kookmin University. His current research focuses on energy-aware computing and IoT middleware. His research interests include service computing, green computing, distributed systems, big data, middleware, IoT, security, and open software.