Design of Raid Controller with Low Power and High Performance

S. Sakena Benazer[1] M. Vidya Shree[2]
[1,2]Assistant Professor
[1,2]ARKA JAIN University, Jharkhand, India

Abstract— This paper presents, an input/output performance of RAID system is increased by a method of partial parity cache (PPC). In PPC the error detection is better than parity cache. The PPC is used to reducing the time taken for estimating the parity bit of a solid-state disk (SSD). SSDs have low complexity, low power consumption and good performance than hard disk drives (HDD). To propose majority logic parity generator architecture in RAID controller. The main objective of this proposed method is to design a low power and low complexity RAID controller for SSD drives and to reduce the parity error rates in RAID controller. To perform read and write operation simultaneously using proposed architecture. This proposed architecture also used to detect and correct the parity errors in SSD.

Keywords: Redundant Array of Inexpensive Disk (RAID), Solid State Disk (SSD), Hard Disk Drive (HDD), Partial Parity Cache (PPC)

I. Introduction

A solid-state disk (SSD) also known as a solid-state drive or electronic disk is a data storage device .Many flash memory chips are there in SSDs. Datas are stored in each flash memory. SSDs have front and back sides. There is no SSD controller in back side of SSD but in front side of SSD has one controller, this is used to control all flash memory chip. Each sides of SSD has an SATA interface, some of the external devices are inserted here. SSDs have no mechanical i.e.,moving components but in hard disk drives (HDD) have an movable read/write heads and spinning disks. This hard disk drives also known as electromechanical magnetic disks or floppy disks. SSDs have lower access time, and low latency compared to Hard disk drives(HDD).

If we bought an ultrabook or ultraportable, we likely had a solid-state drive (SSD) as the primary drive. Every other desktop or laptop had a hard disk drive (HDD).HDD is the nonvolatile storage memory. That is,if the system is off condition that retain the data on system memory. Hard disk drives are having a metal platters with a magnetic coating. Our datas are stored in this magnetic coating. An SSD doing the same function as an HDD, but in SSD does not having any magnetic coatiog, but it having a flash memory chip. The datas are stored in this flash memory chip. That also retain the data when the system is off.

In smartphones and laptops, the flash memory is an important storage device in recent years. Because it consumes low power, gives fast performance and has a better shock resistance compared to hard disk drives. But it has high cost. Flash memory has some characteristics such as, the read and write operations unit is a page, but erase operation unit is a block. Then the speed in different operations are different[1].In a recent flash chip, the erase block size is 128 Kbytes, then the write unit is a word or a page of 2 Kbytes[9]. The small amount of datas are merged using a nonvolatile RAM such as a battery backed up SRAM. But it is a pseudo nonvolatile RAM that needs to be continuously recharged in order to ensure data retention[9]. A Redundant Array of Inexpensive Disks (RAID) architecture is essential to flash memory SSD in order to achiving a high performance and reliability. This can be used to reduce the parity updating cost. Partial parity technique is used to reduce the number of read operations. It can be required to calculate a parity[2].

Compared to conventional hard disks a Random Access First Cache Management method have a better performance, providing larger capacity and lower cost than DRAM. A Random Access First cache method is a hybrid storage architecture that combines both of an SSD based disk cache and a disk drive subsystem. While improving system performance through providing priority to caching random-access data a RAF focuses on extending the lifetime of SSD.RAF splits flash cache into read and write cache to service read/write requests respectively. Here the Read cache only holds random-access data to reduce flash wear and write hits. Then the Write cache performs as a circular write-through log so as to improve system response time. Similar to read cache, write cache only caches random-access data and flushes them to hard disks immediately [4].

Generally a flash memory has many characteristics such as good performance, non-volatility, consumes low power and shock resistance, it is used as a storage media in the embedded and computer system. But some of the disadvantages are also there in flash memory such as high I/O latency due to erase-before-write and poor durability. A RAID technique is used to overcome these problems. In this RAID technology, the blocks are easily detected and corrected so that the reliability is high [6]. In reliability and performance enhancement technique a RAID system is used, it is based on Solid state disk. First, the existing RAID mechanism in the environment of SSD array is analyzed and then develop a new RAID methodology adaptable to SSD array storage system .A solid state disk has better I/O performance compared to hard disk drives. So it becomes more popular in the consumer electronics market [7].A solid state disk’s performance as a function of concurrency, bandwidth, system organization and device architecture [8]. A Hybrid Parity-based Disk Array architecture is a combination of more solid state disks and two hard disk drives. This HPDA is used to improve the performance and reliability of SSD-based storage systems. The reliability of HPDA is better than that of either HDD-based or SSD-based disk array [5].A hybrid approach to large SSDs that combines MLC NAND flash and SLC NAND flash architecture. Each of these flash architectures has its advantages and disadvantages [3].

II. Related work

The RAID system is used in Banking and cloud centres.

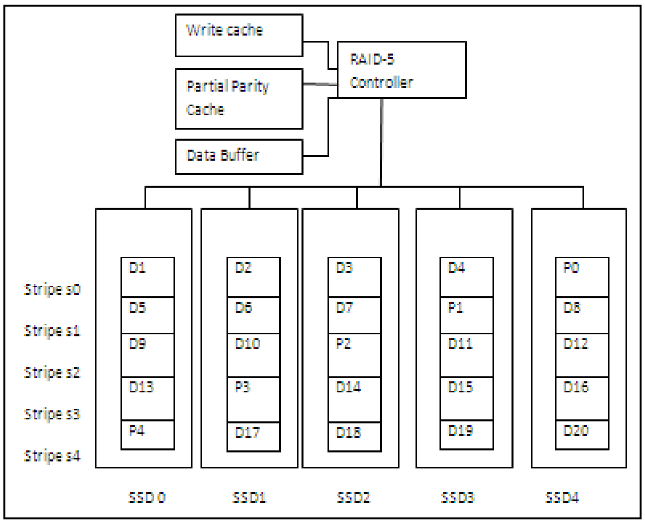


Fig. 1: RAID-5 architecture

Several types of RAID systems are there. RAID 0,RAID 1,RAID 2,RAID 3and RAID 4 but it has some disadvantages, so we have to use RAID 5 system.

The fig.1 shows RAID-5 architecture. This RAID 5 system consists of partial parity cache, data buffer, write cache, RAID 5 controller and Solid State Disks. The RAID 5 system is splitted in to 4+1,that is n+1,here 'n' means number of data storage and '1' represents parity storage. Datas are written in write cache corresponding partial parity bits are generated from partial parity cache. Then the RAID 5 controller controls all SSD. Each SSD has some datas and corresponding parity. It is represented by a Stripe Number, that is S0,S1,S2,….The datas are used to generating the parity using Ex-Or operator. The read operations from SSDs are reduced using data buffer. So the processing speed of system can be increased. Generally there are two conditions in RAID architecture.(i)when SSD in normal condition,(ii)when SSD in busy condition.

(WC-Write Cache,RC-RAID 5 Controller,PPC-Partial Parity Cache,DB-Data Buffer,SSD-Solid State Disk)

A. *When SSD in Normal Condition:*

Under the normal condition of SSD, the host system send the datas to the destination that is SSD via WC, PPC, RC. These datas are written in write cache then it goes to RAID 5 controller. The partial parity bits are generated from partial parity cache, then related data goes to controller. The RAID 5 controller produces the datas and related parity to the corresponding solid state disks. This controller is used to control all disks.

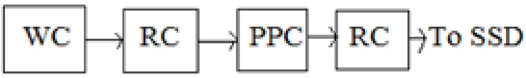


Fig. 2: SSD in normal condition

B. *When SSD in Busy Condition:*

Under the busy condition of SSD, the host system send the datas to the destination that is SSD via WC, DB, PPC, RC. These datas are written in write cache then it goes to RAID 5 controller.

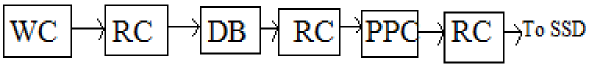


Fig. 3: SSD in busy condition

Then the datas are stored in data buffer. Again it goes to controller. The partial parity bits are generated from partial parity cache, then related data goes to controller. The RAID 5 controller produces the datas and related parity to the corresponding solid state disks.

III. RAID 5 CONTROLLER

Fig.4 shows the RAID 5 Controller design. The RAID 5 controller consists of

1) Constant amplitude encoder
2) Multi code generator
3) Orthogonal multiplexer
4) Orthogonal multiplier

The input of RAID 5 controller has 7bits.These inputs are given to the multicode generator. The multicode generator generates this 7bits into 9bits.Then finally we get 4bits as an output of RAID 5 controller.

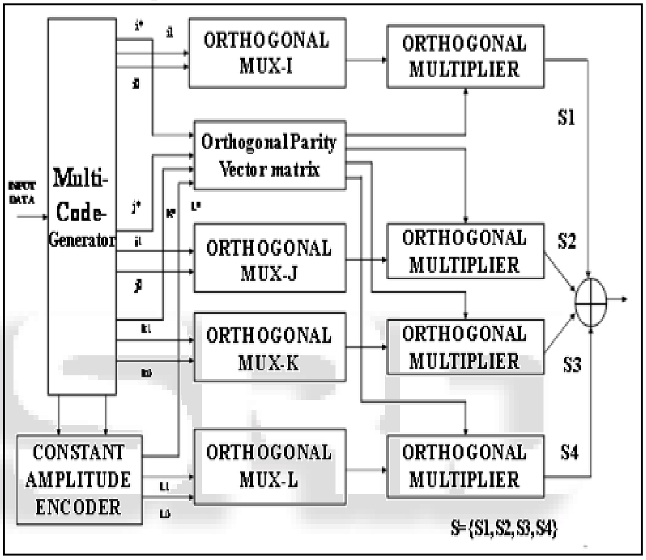


Fig. 4: RAID Controller design

A. *Constant Amplitude Encoder:*

In this work, three bits (L*,L1,L0) are generated from three groups of parallel bits (i*,i1,i0),(j*,j1,j0),(k*,k1,k0) According to the following equation

L*= ~(i* ^ j* ^k*)
L1= i1 ^j1 ^k1
L0= i0 ^j0 ^k0

B. *Multi Code Generator:*

The multi code generator consists of 2-blocks.

- Serial-to-Parallel Converter.
- Gold Sequence Generator.

1) *Serial-to-Parallel Converter:*

The serial to parallel converter is used to converts the data bits in to number of branches according to the length of Gold Sequence.

2) *Gold Sequence Generator:*

There are two pseudo random noise sequences generating the gold sequence using Ex-or operator. The formula for pseudo random noise sequence is

$$\text{PN Sequence}=2^{m}-1$$

Where 'm' is the no.of flipflops

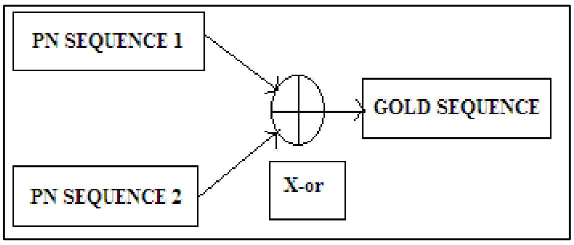


Fig. 5: Gold Sequence generator

C. *Orthogonal Multiplexer:*

Generally four orthogonal muxes are there in RAID 5 controller. That is I,J,K and L.The bits ‘00’ are inbuilted in I, bits ‘01’ are inbuilted in J, bits ‘10’ are inbuilted in K and bits ‘11’ are inbuilted in L.

D. *Orthogonal Multiplier:*

$$S1=b\begin{bmatrix} Ci \\ Cj \\ Ck \\ Cl \end{bmatrix} =[i^* \ k^* \ j^* \ l^*]\begin{bmatrix} Ci \\ Cj \\ Ck \\ Cl \end{bmatrix}$$

$=i^*.Ci + j^*.Cj + k^*.Ck + l^*.Cl$

Where,’b’represents Orthogonal Parity Vector Matrix and ‘C’ is the Walsh-Hadamard Matrix.

IV. Partial Parity Cache Architecture

Fig.6 shows the Partial Parity Cache Architecture. It has four blocks. Each blocks consists of 7bits.The PPC has two multiplexers.

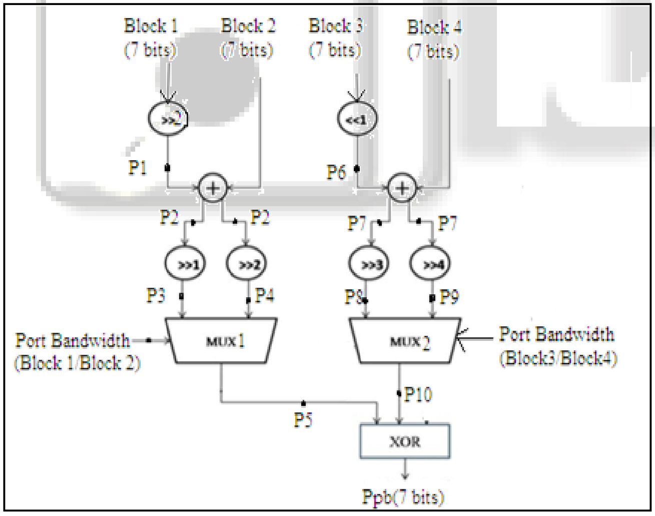


Fig. 6: Partial parity cache architecture

In mux1, the blocks 1 and blocks 2 are divided then blocks 3 and blocks 4 are divided in mux 2. In block 1 the 7 bits are 2 times right shifted, then it can produce P1.This P1 is added with 7 bits from block 2.These can produce P2.This P2 is one time right shifted then it can produce P3,again P2 is two times right shifted then it can produce P4.The mux 1 has two inputs that are P3 and P4. P5 is the output of mux 1. The 7 bits are 1 time left shifted in block 3, then it can produce P6.This P6 is added with 7 bits from block 4.These can produce P7.

This P7 is three times right shifted then it can produce P8, again P7 is four times right shifted then it can produce P9.This P8 and P9 given as an input of mux 2.The output of mux 2 is P10. This P5 and P10 can produce partial parity bit.This ppb has 7 bits.

V. Proposed Work

A. *Proposed RAID 5 System Architecture:*

The fig.7 shows Proposed RAID 5 system.

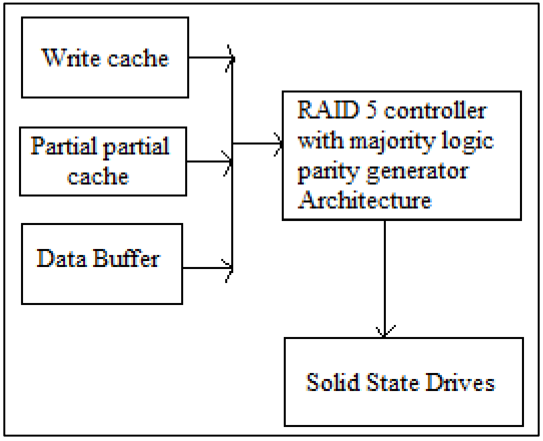

Write cache
Partial partial cache
Data Buffer
RAID 5 controller with majority logic parity generator Architecture
Solid State Drives

Fig. 7: Proposed RAID 5 system

Here the majority logic parity generator architecture in RAID 5 controller to reduce the power consumption and improves the latency performance. The proposed architecture detects and corrects the parity errors in SSD drives.

B. *Proposed ML Parity Generator architecture:*

The Fig.8 shows the majority logic Parity Generator architecture.

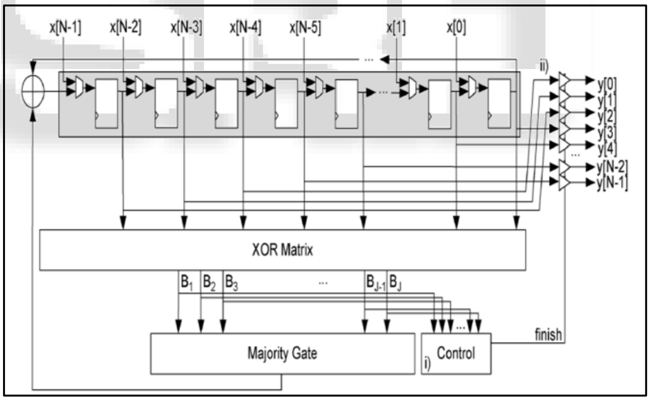


Fig. 8: Proposed ML Parity Generator

The Proposed ML Parity Generator architecture consists of Cyclic shift register, XOR matrix, Majority gate, XOR for correction, Control unit and Output tri state buffers. The Additional hardware to perform the error detection are control unit and output tris tate buffers. The control unit which triggers a finish flag when no errors are detected. The tri state buffer outputs are always in high impedance unless the control unit sends the finish signal so that the current values of the shift register are forwarded to the output ‘y’.

C. *Control Unit:*

The fig.9 shows control unit. A counter is used in control unit. In first three iterations, the control unit evaluates the {Bj}. When the result is “0,” the FSM sends out the finish signal indicating that the processed word is error-free.

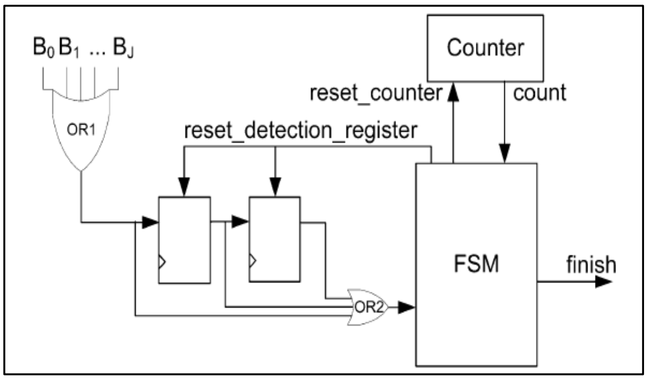


Fig. 9: Control Unit

In the other case, if the result is "1," the ML parity generator process runs until the end.

VI. RESULTS AND DISCUSSION

A. Simulation Result for PPC:

Fig.10 shows that simulation result of partial parity cache. Four packets are there,each packet have 24 bits.In this packet first 4bits represents transaction id,next 4bits represents address and last 4bits represents data.Then the output of ppc has consists of 7bits.

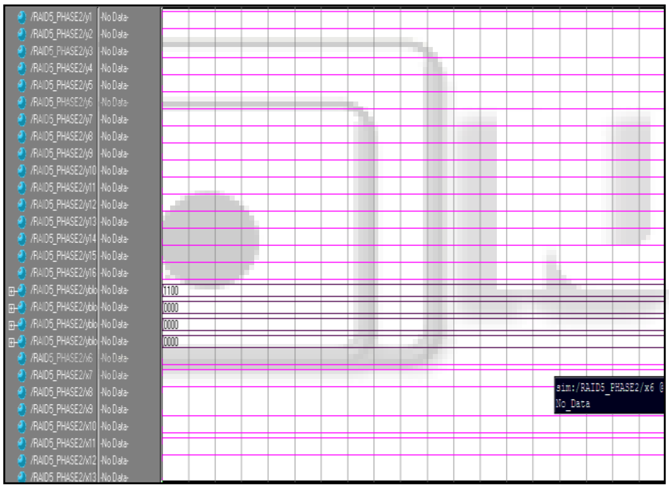

/RAID5_PHASE2/y1 -No Data-
/RAID5_PHASE2/y2 -No Data-
/RAID5_PHASE2/y3 -No Data-
/RAID5_PHASE2/y4 -No Data-
/RAID5_PHASE2/y5 -No Data-
/RAID5_PHASE2/y6 -No Data-
/RAID5_PHASE2/y7 -No Data-
/RAID5_PHASE2/y8 -No Data-
/RAID5_PHASE2/y9 -No Data-
/RAID5_PHASE2/y10 -No Data-
/RAID5_PHASE2/y11 -No Data-
/RAID5_PHASE2/y12 -No Data-
/RAID5_PHASE2/y13 -No Data-
/RAID5_PHASE2/y14 -No Data-
/RAID5_PHASE2/y15 -No Data-
/RAID5_PHASE2/y16 -No Data-
/RAID5_PHASE2/yblo -No Data- 1100
/RAID5_PHASE2/yblo -No Data- 0000
/RAID5_PHASE2/yblo -No Data- 0000
/RAID5_PHASE2/yblo -No Data- 0000
/RAID5_PHASE2/x6 -No Data-
/RAID5_PHASE2/x7 -No Data-
/RAID5_PHASE2/x8 -No Data-
/RAID5_PHASE2/x9 -No Data-
/RAID5_PHASE2/x10 -No Data-
/RAID5_PHASE2/x11 -No Data-
/RAID5_PHASE2/x12 -No Data-
/RAID5_PHASE2/x13 -No Data-
sim:/RAID5_PHASE2/x6
No_Data

Fig. 10: Simulation Result for PPC

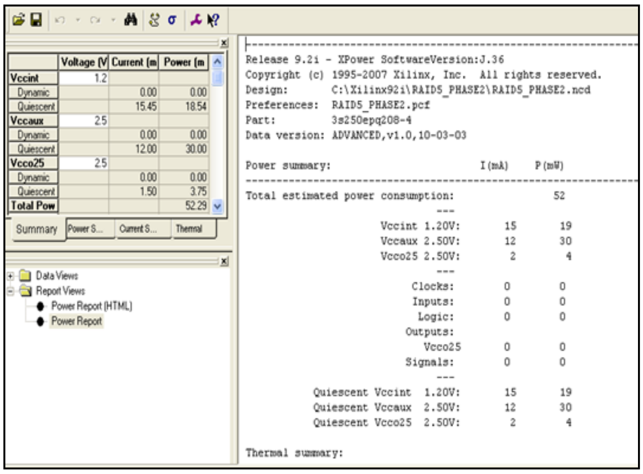
B. Performance Analysis:

To analysis the performance of power consumption, current consumption and latency.

Parameters	Existing Method	Proposed Method
Power Consumption	81mW	52mW
Current Consumption	46mA	29mA
Latency	6.209ns	5.667ns

Table 1: List of Parameter with T=the Comparison of Existing Method and Proposed Method

C. Power Consumption:



	Voltage (V	Current (m	Power (m
Vccint	1.2		
Dynamic		0.00	0.00
Quiescent		15.45	18.54
Vccaux	2.5		
Dynamic		0.00	0.00
Quiescent		12.00	30.00
Vcco25	2.5		
Dynamic		0.00	0.00
Quiescent		1.50	3.75
Total Pow			52.29

```
--------------------------------------------------------------------------
Release 9.2i - XPower SoftwareVersion:J.36
Copyright (c) 1995-2007 Xilinx, Inc.  All rights reserved.
Design:       C:\Xilinx92i\RAID5_PHASE2\RAID5_PHASE2.ncd
Preferences:  RAID5_PHASE2.pcf
Part:         3s250epq208-4
Data version: ADVANCED,v1.0,10-03-03

Power summary:                           I(mA)     P(mW)
--------------------------------------------------------------------------
Total estimated power consumption:                   52
                                   ---
                      Vccint 1.20V:        15        19
                      Vccaux 2.50V:        12        30
                      Vcco25 2.50V:         2         4
                                   ---
                            Clocks:         0         0
                            Inputs:         0         0
                             Logic:         0         0
                           Outputs:
                             Vcco25         0         0
                           Signals:         0         0
                                   ---
              Quiescent Vccint  1.20V:     15        19
              Quiescent Vccaux  2.50V:     12        30
              Quiescent Vcco25  2.50V:      2         4

Thermal summary:
```

VII. CONCLUSION

In this paper, the performance of RAID system is increased by a method of Partial Parity Cache. We proposed a low power and low complexity RAID 5 controller for SSD drives and a majority logic parity generator architecture in RAID 5 controller that performs read and write operations simultaneously. The proposed architecture detects and corrects the parity errors in SSD drives. The power consumption and parity error rates are reduced.

REFERENCES

[1] C.-C. Chung, D. Sheng, and N.-M. Hsueh, "A high-performance wearleveling algorithm for flash memory system," IEICE Electron. Exp.,vol. 9, no. 24, pp. 1874–1880, Dec. 2012.

[2] S. Im and D. Shin, "Flash-aware RAID Techniques for dependable and high-performance flash memory SSD," IEEE Trans. Comput., vol. 60,no. 1, pp. 80–92, Jan. 2011.

[3] L.-P. Chang, "A hybrid approach to NAND-flash-based solid-state disks," IEEE Trans. Comput., vol. 59, no. 10, pp. 1337–1349, Oct. 2010.

[4] Y. Liu, J. Huang, C. Xie, and Q. Cao, "RAF: A random access first cache management to improve SSD-based disk cache," in Proc. NAS, Jul. 2010, pp. 492–500.

[5] B. Mao, H. Jiang, D. Feng, S. Wu, J. Chen, L. Zeng, and L. Tian, "HPDA: A hybrid parity-based disk array for enhanced performance and reliability," in Proc. IEEE IPDPS, Apr. 2010, pp. 1–12.

[6] Y. Lee, S. Jung, and Y. H. Song, "FRA: A flash-aware redundancy Array of flash storage devices," in Proc. Hardw./Softw. Codesign Syst. Synthesis, Oct. 2009, pp. 163–172.

[7] K. Park, D.-H. Lee, Y. Woo, G. Lee, J.-H. Lee, and D.-H. Kim, "Reliability and performance enhancement technique for SSD array storage system using RAID mechanism," in Proc. ISCIT, Sep. 2009, pp. 140–145.

[8] R. Ho, K. W. Mai, and M. A. Horowitz, "The performance of PC solid state disks (SSDs) as a function of bandwidth, concurrency, device architecture, and system organization," in Proc. ACM/IEEE ISCA, Jun. 2009,pp. 279–289.

[9] C. Lee, S. H. Baek, and K. H. Park, “A hybrid flash file system based on NOR and NAND flash memories for embedded devices,” IEEE Trans. Comput., vol. 57, no. 7, pp. 1002–1008, Jul. 2008.

[10] N. Agrawal, V. Prabhakaran, T. Wobber, J. D. Dacis, M. Manasse, and R. Panigrahy, “Design tradeoffs for SSD performance,” in Proc. USENIX ATC, Jun. 2008, pp. 57–70.

[11] J.H. Yoon, E.H. Nam, Y.J. Seong, H.S. Kim, B.S. Kim, S.L. Min, and Y.K. Cho, “Chameleon: A High Performance Flash/FRAM Hybrid Solid-Sate Disk Architecture,” IEEE Computer Architecture Letters, vol. 7, no. 1, pp. 17-20, Jan.-June 2008.

[12] L.P. Chang, “Hybrid Solid-State Disks: Combining Heterogeneous NAND Flash in Large SSDs,” Proc. 13th Asia and South Pacific Design Automation Conf., 2008.

[13] J. G. Elerath and M. Pecht.” Enhanced Reliability Modeling of RAID Storage Systems”. In Proceedings of 37th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN’07), Jun. 2007.

[14] S.H. Lim and K.H. Park, “An Efficient NAND Flash File System for Flash Memory Storage,” IEEE Trans. Computers, vol. 55, no. 7, pp. 906-912, July 2006.