The Design and Performance Study of a Multi-heads Parallel Access Disk

Kuo-An Hwang
Professor, Department of Computer Science and Information Engineering & Graduate Institute of Networking and Communication Engineering, Chaoyang University of Technology
kahwang@cyut.edu.tw

Chi-Yi Kuo
Graduate student, Department of Computer Science and Information Engineering, Chaoyang University of Technology
s9527643@cyut.edu.tw

Chia-Hao Yang
Graduate Institute of Informatics, Doctoral program, Chaoyang University of Technology
s9433901@cyut.edu.tw

摘要

傳統上使用者對磁碟裝置存取資料時，裝置內磁頭移動至指定位置進行存取作業，由於附有磁頭的讀寫臂移動至磁柱時，僅對使用者所需資料磁區執行存取作業，其他磁頭則進入閒置狀態。本研究提出以多磁頭同步並行存取磁柱的方法，減少磁頭閒置狀態來縮短磁碟裝置存取作業時間，並以此方法在單一磁碟中建構 RAID 磁碟陣列，解決其中裝置不完全同步問題。最後探討以此方式存取作業的磁碟裝置的存取效能，並與傳統磁碟以及存取效能最高 RAID-0 做效能比較。

關鍵詞：磁碟陣列、同步並行存取、多磁頭。

Abstract

Traditionally, when a user accesses the data on a disk device, the R/W arm in the disk device moves to the designated position to carry out the access operation; then the access operation is carried out only by the R/W head to the requested data sector. Other R/W heads enter the idle condition. The research proposes an approach, the Multiple Heads Parallel Access to Disk cylinder, to improve the time of access operation of the disk device. It can be used to construct RAID-0 disk array in a single disk with fully synchronous access. Furthermore, we explore the access performance of disk device in this way and make comparison with the traditional disk and the RAID-0 disk array.

Keywords: Disk Array, Synchronous Parallel Access, Multiple Magnetic Heads Process

1. Introduction

The disk is an important storage device of the computer nowadays, which affects the execution efficiency of the computer. Most of application programs and their data are stored in the disk device [2, 5, 16, 18]. It takes a large proportion of the overall execution time to read data out of and write data into the disk, therefore, the higher the access efficiency of the disk is, the shorter the execution time will be, thus the processing speed of the computer is accelerated [1, 13, 16].

The key factors that affect the efficiency of the access operation of the disk device most are the Seeking and rotation latency. In respect of the improvement of the seek operation, Robbins presented and compared the track-seeking methods [14, 16] of various disk heads under various circumstances. He found out track-seeking types that are correspondingly suitable to certain situation(s) so as to optimize and shorten the seeking time. Frank analyzed the time-sharing utilization of the disk device, and the correlation between the system and the seek operation and carried out the optimization [4]. Yen, through the configuration of data by the systematic planning of various file systems or file grouping [20], etc., reduced the frequency of seek operation[16,20].

When redundant data is accessed, data on disk sectors are accessed by the frequent seek operation of the disk heads, and each seek action will give rise to the rotation latency. Despite that the rotation latency costs can be improved by increasing the rotation speed of the spindle motor. The increase of the rotation speed will severely affect the density of stored data.

Furthermore, the read/write hardware design and the high rotation speed will cause vibration and noise, etc. Therefore, the rotation speed of the spindle motor can not be increased all the time.

With the data quantity getting larger and larger, the size of the disk device is becoming smaller and smaller, and the price is cheaper and cheaper day by day [8]. Multiple disk devices have been constructed to handle works together based on the concept of multi-processing. David A Patterson, Garth Gibson and Randy H Katz put forward different RAID architectures [9, 10]. RAID-0 divides a piece of data into many blocks which can be accessed from many disks, thus enables the highest access efficiency among all disk arrays [3]. Other RAID architectures are aimed to enhance the reliability of stored data.

Even through the data access efficiency of the disk device can be enhanced by RAID, but due to the independent operation of each disk device, the access operating of the disk array can not able to achieve fully synchronous. Moreover, only one of the R/W head is used in an access operation in each disk device, the improvement of the access performance is limited.

2. Multi-heads parallel access disk

A multi-heads parallel access disk (MHPA) is a disk in which all the R/W heads of the disk are used to access data concurrently. Each disk head inside the disk driver is provided with an exclusive processing path.

2.1 Fully parallel architecture in a MHPA disk, FP-MHPA

The data stored in a MHPA disk can be arranged as a fully parallel architecture, in which, each sector of data are divided into all the surface of the dusk. As show in Figure 1, when the writing-in operation is carried out, the data is divided by the Data Segmentation Unit. The divided data is sent to all the disk heads for synchronous writing-in operation. The data read by each R/W head are merged into the desired data block of the read operation.

This approach can realize the synchronous parallel work environment so as to improve the efficiency of the disk access. The blocks of data to be accessed by each disk are divided into many clusters according to the number of R/W heads. The quantity of data to be accessed by each R/W head is 1/m of that to be accessed by the device for a MHPA disk with m R/W heads.

As shown in Figure 2, data parallel access is carried out in a disk driver with 4 disk heads in a segment size of 512Bytes. FP-MHSPA is adopted so all disk heads are engaged in the access operation; each disk head is in charge of 128Bytes of data, and the data access time spent will be reduced by 1/4 times. The number of sectors in a track is 4 times of the traditional disk.

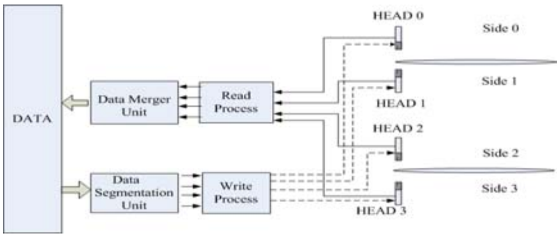

Fig. 1 MHPA R/W Process

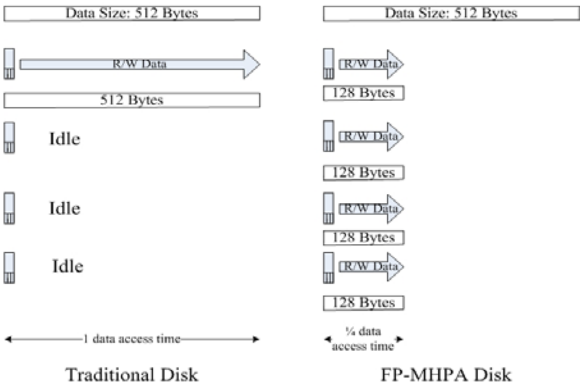

Fig. 2 Traditional Disk & FP- MHPA Disk access 512bytes

2.2 Sector-wise parallel architecture in a MHPA disk, SP-MHPA

In a SP-MHPA disk, data are stored in sector-wise. That is, as shown in Fig 3, a traditional sector of data are stored in a sector of SP-MHPA disk, and the subsequent sector is stored in next surface of the disk, etc.

For example, if a RAID-0 is constructed by four disk devices, as shown in Figure 3, it can be constructed in a MHPA disk with four R/W heads. In the figure, it is shown with 32 blocks of data allocated. At each access operation, 4 blocks of data are acceded concurrently. As shown in Figure 4, the traditional disks build a RAID-0 with four disks.

A SP-MHPA disk is functional like a RAID-0 if we view a surface of SP-MHPA disk as a disk of the RAID. Thus the RAID structure in a MHPA disk provides fully synchronous access operations.

If the parallel access mechanism is disabled, then the SP-MHPA disk is exactly compatible with a traditional disk.

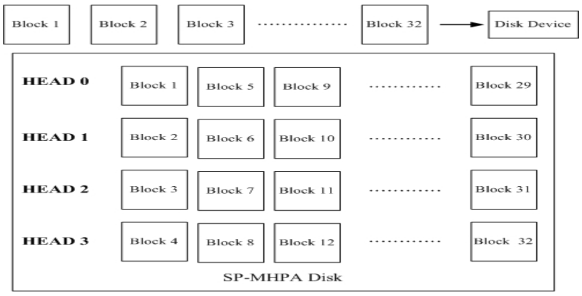

Fig.3 SP-MHPA

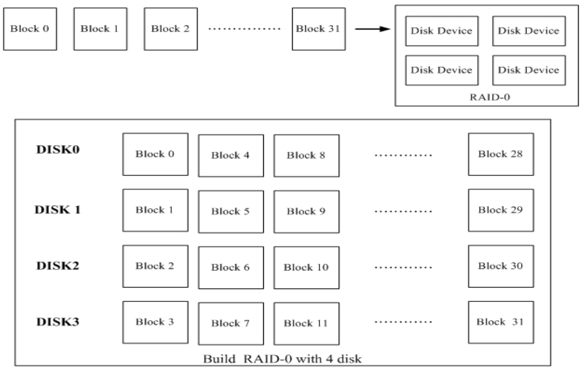

Fig.4 Build RAID-0 with 4 Disks

2.3 Build MHPA Disks of RAID-0

MHPA can be used to construct various different RAID architectures. RAID-0 can be implemented with even number of disk drivers. For example, if a RAID-0 is constructed by four disk devices with 4 disk heads for each; the disk driver adopts MHPA approach to plan the data access, as shown in Figure 5.

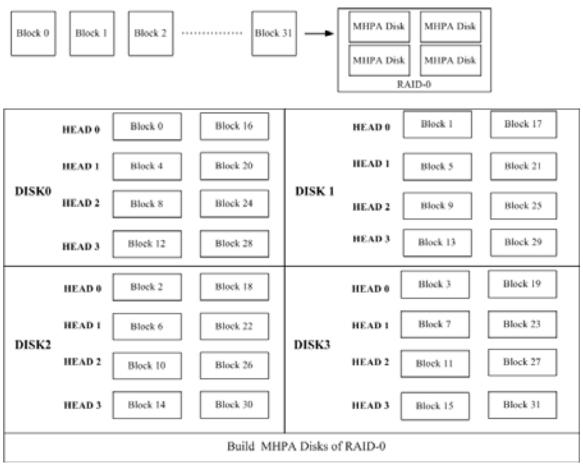

Fig.5 Build MHPA Disks of RAID-0

3. Performance Evaluation

To study the access performance of the MHPA disk, we made a comparison between the approach of MHPA disk and traditional disk. The access time of each approach is calculated with respect to the following parameters:

Sk_{Avg} : Average seeking time

Rs : Rotation speed

Se : The number of R/W heads of a disk

Ts : The number of sectors of a disk track

N : The number of disks of a RAID

Where $R_{Avg} = 1/2Rs$

Let

Ds Denotes the Total number of sectors data to be accessed.

Dt Denotes the Total access time

Ss Denotes the number of tracks accessed.

Cs Denotes the number of seek operations.

Td Denotes the data transfer time for a sector of data.

Then

$$Dt = Cs(S_{Avg}) + Ss(R_{Avg}) + (Ds * Td)$$

3.1 Access time of a Traditional disk

When a traditional disk driver is started, refers to the time it takes to seek and select a disk head, and then wait for the disk to rotate to the designated position to start the access operation. When the data access involves many disk surfaces, Ss is adopted to represent the times of shifting. The access time of traditional disk driver is as shown in Equation 1.

$$Cs = \lceil Ds/(Ts * Se) \rceil$$

$$Ss = \lceil Ds/Ts \rceil$$

$$Td = 1/(Ts * Rs)$$

$$Dt = (Cs * Sk_{Avg}) + (Ss * R_{Avg}) + (Ds/(Ts * Rs)) \quad \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

Where $\lceil Ds/Ts \rceil$ refers to the minimum integer that is larger than or equal to D_S/T_S, it is the number of tracks to be accessed. $\lceil Ds/(Ts * Se) \rceil$ is the number of the cylinders to be accessed.

3.2 Access time of a RAID-0

At least two disk drivers are required to construct a RAID-0. Data to be processed are divided into parts of the same quantity of the devices. The access performance of a RAID-0 is optimal if all the disks are fully synchronized, as shown in Equation 2.

$$Cs = \lceil (Ds/N)/(Ts*Se) \rceil$$
$$Ss = \lceil ((Ds/N)/Ts) \rceil$$
$$Td = 1/(Ts*Rs)$$
$$Dt_{Avg} = (Cs*Sk_{Avg}) + (Ss*R_{Avg}) + (\lceil Ds/N \rceil/(Ts*Rs)) \quad \text{………………. (2)}$$

Where N sectors can be accessed at the same time

3.3 Access time of a FP-MHPA disk

FP-MHPA is adopted so all disk heads are engaged in the access operation. The access time of FP-DISK is as shown in Equation 3.

$$Cs = \lceil Ds/(Ts*Se) \rceil$$
$$Td = 1/(Ts*Rs*Se)$$
$$Dt = (Cs*(Sk_{Avg} + R_{Avg})) + (Ds/(Ts*Rs*Se)) \quad \text{………………... (3)}$$

If the workload when the number of sector to be accessed is m, m < N, it needs one traditional sector access time for RAID-0, In FP-DISK, it needs only m/N of one traditional sector access time.

3.4 Access time of a SP-MHPA disk

As the SP-MHPA Disk adopts the shifting of the disk cylinder instead of the shifting of the disk surfaces, therefore, Cs here is used to represent the shifting time of the disk cylinder. The access time is as shown in Equation 4.

$$Cs = \lceil Ds/(Ts*Se) \rceil$$
$$Td = 1/(Ts*Rs*Se)$$
$$Dt = Cs(Sk_{Avg} + R_{Avg}) + (\lceil Ds/Se \rceil/(Ts*Rs)) \quad \text{………………. (4)}$$

3.5 Access time of a MHPA RAID-0

A RAID-0 is constructed with the MHPA Disk. The total access time is as shown in Equation 5.

$$Cs = \lceil (Ds/N)/(Ts*Se) \rceil$$
$$Td = 1/(Ts*Rs*Se)$$
$$Dt_{Avg} = Cs(Sk_{Avg} + R_{Avg}) + (\lceil Ds/N \rceil/(Ts*Rs*Se)) \quad \text{……….…. (5)}$$

3.6 Performance Comparison

To make a more clear sense of the performance comparison, a workload with 1000 sectors of data in sequential access mode is used as the sample of the study. The parameters of the disk are assumed as the followings:

Rs = 7200RPM,
Sk_{Avg} =12ms
R_{Avg} =4.17ms
Se =4
Ts =144

If the access is in random access mode, Ds is taken as 1 sector.

The total access times are calculated for both of the continuous access mode and the random access mode. The comparison between the access times for traditional disk, FP-MHPA disk and SP-MHPA disk are shown in Table 1. While that of traditional RAID-0 with four disks and RAID-0 with four MHPA disks are shown in Table 2.

Table 1: Traditional disk, FP-MHPA Disk and SP-MHPA Disk

Access Mode	Continuous Access	Random Access
Traditional disk	111.06 ms	16.228 ms
FP-MHPA Disk	46.80 ms	16.184 ms
SP-MHPA Disk	46.80ms	16.228 ms

Table 2: Traditional RAID-0 vs. MHPA RAID-0

Access Mode	Continuous Access	Random Access
Traditional RAID-0 with 4 Disks	Access time $\geq$ 34.81 ms	16.228 ms
MHPA RAID-0 with 4 disks	Access time $\geq$ 19.79 ms	16.228 ms

4. Conclusion

From the efficiency evaluation, we know that the MHPA disk can improve the disk access speed. It can make multiple disk heads free of idle condition. Furthermore, it can solve the mismatch problem between the internal transmission speed and the external transmission speed, thus improves the access efficiency of the single disk device.

Disk Arrays are constructed with multiple disk devices in the expectation of that the synchronous data access of the devices can be achieved. However, the fully synchronous data access of devices does not realize. Therefore, our MHPA approach is adopted to construct a Disk Array in a single disk driver. It can solve the problem of delay caused by the non-fully synchronous access of disk devices in the Disk Array. The construction of a Disk Array with MHPA disk drivers can get more improvement to the access efficiency of the Disk Array.

The speeds of the CPU and the RAM have been substantially increased; however, the growth of the efficiency of the I/O disk device falls far behind that of the CPU and RAM. Therefore, the MHPA can be adopted to improve the work efficiency and to reduce the idle conditions of the CPU and RAM, so as to improve the overall processing efficiency of the computer.

Reference

[1] Bachman, C. W., "The evolution of storage structures," ***Communications of ACM,*** Vol. 15, Issue 7, pp.628-634, 1972.

[2] Bass, J. E., "A Peripheral-Oriented Microcomputer System," ***Proceedings of the IEEE***, Vol. 64, No. 6, 1976.

[3] Chen, P. M., and Patterson, D. A., "Maximizing performance in a Striped Disk Arrays," ***Proceedings of 17th Annual International Symposium on Computer Architecture***, pp. 28-31, 1990.

[4] Frank, H., "Analysis and optimization of disk storage devices for time-sharing systems," ***Journal of the ACM,*** Vol. 16, Issue 4, pp. 602-620, 1969.

[5] Golden, D., and Pechura, M., "The Structure of microcomputer file systems," ***Communications of the ACM,*** Vol. 29, Issue 3, pp. 222-230, 1986.

[6] Jeppesen, J., Allen, W., Anderson, S., and Pilsl, M., "Hard Disk Controller: the Disk Drive's Brain and Body," ***Proceedings of 2001 International Conference on Computer Design***, pp. 262-267, 2001.

[7] Jin, H., and Hwang, K., "Stripped mirroring RAID architecture," ***Journal of Systems Architecture***, Vol. 46, Issue 6, pp. 543-550, 2000.

[8] Jing, G., Xu, H., and Pan, L., "Implementation of EIDE Disk arrays System for Mass Data Backup," ***IEEE Aerospace and Electronic Systems Magazine***, Vol. 19, Issue 11. pp. 24-29, 2004.

[9] Katz, R. H., Gibson, G. A., and Patterson, D. A., "Disk System Architecture for High Performance Computing," ***Proceedings of the IEEE***, Vol. 77, No. 12, pp. 1842-1858, 1989.

[10] Katz, R. H., Gibson, G. A., and Patterson, D. A., "Introduction to Redundant Arrays of Inexpensive Disks (RAID)," ***IEEE 34th International Conference of Computer Society***, pp. 112-117, 1989.

[11] Mishra, S. K., and Mohapatra, P., "Performance Study of RAID-5 Disk Arrays with Data and Parity Cache," ***International Conference on Parallel Processing***, Vol. 1, pp. 222-229, 1996.

[12] Moreno, L., Roca, I., and Luque, E., "Floppy disk controller for a computer architecture course," ***IEEE Transactions on Education***, Vol. 32, Issue 2, pp. 112 – 117, 1989.

[13] Patterson, D. A., and Hennessy, J. L., ***Computer Organization & Design The Hardware/Software Interface,*** Morgan Kaufmann, Publishers, inc, 2000.

[14] Robbins, S., "A disk head scheduling simulator," ***35th technical symposium on Computer science education***, Session: Operating system, pp. 325-329, 2004.

[15] Rotem, D., "Analysis of Disk Arm Movement for Large Sequential Reads," ***Proceedings of the 8th ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems***, pp. 47-54, 1992.

[16] Silberschatz, A., Galvin, P. B., and Gagne, G., ***Operating System Concepts, Windows XP update***, 6th ed., John Wiley & Sons, Inc, 2003.

[17] Sumi, M., Ohhashi, M., Yanagi, H., Kaizuka, M., Shimizu, H., and Maruyama, T., "A Hard Disk Channel Processor," ***IEEE Journal of Solid-State Circuits,*** Vol.20, Issue 3, pp. 787-792, 1985.

[18] Tuohy, S. L., "The logical format of a floppy disk," ***Journal of Computer Sciences in colleges***, Vol.17, Issue 3, 2002.

[19] Worthington, B. L., Ganger, G. R., and Patt, Y. N., "Scheduling Algorithms for Modern Disk Devices," ***ACM SIGMETRICS conference on Measurement and modeling of computer systems***, pp. 241-251, 1994.

[20] Yen, T., Arul, J., Wu, J. S., Chan, I. F., and Tan, K. H., "Using Files Grouping to Improve the Disk Performance," ***2006 15th IEEE International Symposium on High Performance Distributed Computing***, pp. 365-366, 2006.